

Títol: Arquitectura per a un simulador de cirurgia endoscòpica
prostàtica

Volum: 1

Alumne: Carlos Villanova Arboledas

Director/Ponent: Robert Joan Arinyo

Departament: LSI

Data: 20 de Gener de 2008

DADES DEL PROJECTE

Títol del Projecte: Arquitectura per a un simulador de cirurgia endoscòpica prostàtica

Nom de l'estudiant: Carlos Villanova Arboledas

Titulació: Enginyeria en Informàtica

Crèdits: 37,5

Director/Ponent: Robert Joan Arinyo

Departament: LSI

MEMBRES DEL TRIBUNAL *(nom i signatura)*

President:

Vocal:

Secretari:

QUALIFICACIÓ

Qualificació numèrica:

Qualificació descriptiva:

Data:

ÍNDEX

1	INTRODUCCIÓ.....	7
1.1	Motivació del projecte.....	7
1.2	Objectius del projecte.....	7
1.3	Gestió de riscos.....	8
2	ANÀLISI: ESPECIFICACIÓ	10
2.1	Requeriments no funcionals.....	10
2.2	Diagrames de casos d'ús	13
2.2.1	Actors	13
2.2.2	Interaccions amb model.....	14
2.2.3	Gestió d'usuaris i fitxers.....	15
2.3	Descripcions dels casos d'ús.....	16
2.3.1	Interaccions amb model.....	16
2.3.2	Gestió d'usuaris i fitxers.....	25
2.4	Diagrames de blocs i d'estat.....	31
2.5	Diagrames de classes.....	34
2.6	Diagrames de seqüència del sistema	36
2.6.1	Interaccions amb model.....	36
2.6.2	Gestió d'usuaris i fitxers.....	40
2.7	Contractes de les operacions del sistema	45
2.7.1	Interaccions amb model.....	45
2.7.2	Gestió d'usuaris i fitxers.....	49
3	ANÀLISI: DISSENY	54
3.1	Arquitectura física del sistema	54
3.2	Arquitectura lògica del sistema	57
3.2.1	Disseny de la capa de presentació.....	58
3.2.2	Disseny de la capa de domini	73
3.2.3	Disseny de la capa d'accés a dades.....	82
4	IMPLEMENTACIÓ.....	85
4.1	Descripció de l'entorn de treball	85
4.2	Tecnologies utilitzades	86
4.2.1	Qt.....	86
5	MANUAL D'USUARI.....	92
5.1	Interaccions amb model	92
5.2	Gestió d'usuaris i fitxers	93
6	ANÀLISI DE LA PLANIFICACIÓ I VALORACIÓ ECONÒMICA	97
6.1	Anàlisi de la planificació.....	97
6.2	Valoració econòmica.....	101
7	CONCLUSIONS.....	103
7.1	Conclusions personals.....	103
7.2	Treball futur	104
8	AGRAÏMENTS	105
9	BIBLIOGRAFIA I REFERÈNCIES	106
9.1	Bibliografia	106
9.2	Referències electròniques.....	106
10	ANNEXOS	107
10.1	Glossari	107

1 Introducció

Aquest és l'apartat introductori de la memòria del projecte en el que s'expliquen els detalls sobre l'origen del projecte i el llistat d'objectius a assolir. També es descriu la importància d'aquest projecte pel camp de la medicina en un futur no massa llunyà. Com afegit, aquesta introducció incorpora una gestió dels riscos del projecte.

1.1 Motivació del projecte

Aquest projecte ve motivat perquè en l'actualitat no existeix cap simulador de cirurgia endoscòpica prostàtica que funcioni amb garanties elevades d'èxit. Això significa que avui en dia els cirurgians no disposen d'una eina informàtica per a poder fer pràctiques en aquests tipus d'operacions. Per tant, avui en dia els cirurgians amb poca experiència es limiten a observar com operen els cirurgians més experimentats per tal de poder aprendre les tècniques precises que comporta aquesta operació. És molt important afirmar que aquestes operacions tenen molts riscos per al pacient ja que qualsevol tall mínimament mal fet pot ser catastròfic. Totes aquestes raons són suficients per a que es vulgui desenvolupar una eina de simulació eficaç. De fet aquesta recerca ha estat promoguda per l'Institut Català d'Urologia i Nefrologia, pel centre de Referència en Enginyeria Biomèdica de Catalunya, per *Red de Grupos del Instituto Carlos III*, per IM³ i pel *Ministerio de Educación y Ciencia*. Això vol dir que hi ha una gran expectació en quan a que es desenvolupi satisfactòriament el simulador.

1.2 Objectius del projecte

L'objectiu principal d'aquest projecte és efectuar l'anàlisi funcional d'un simulador computeritzat per a l'entrenament en cirurgia endoscòpica de la pròstata. La idea bàsica és que els cirurgians puguin obtenir un simulador el més real i ràpid possible. El projecte vist de forma global es subdivideix en dos subprojectes, un d'ells és aquest i l'altre correspondrà a la implementació completa del simulador, tenint com a referència tot el treball realitzat en aquest projecte.

Com sabreu, les operacions de pròstata han de ser molt precises i, per tant, es necessita disposar d'un simulador que estigui a la alçada i que realment sigui útil per fer un intens

aprenentatge abans de fer les operacions reals. Ara mateix existeix un petit prototip de simulador d'operacions prostàtiques però conté nombroses deficiències funcionals, és per això que es requereix efectuar una anàlisi de l'aplicació i dissenyar una arquitectura viable. El principal problema del simulador actual és la lentitud en el procés de tall de la massa prostàtica. Això és degut a la gran quantitat de càlculs que realitza l'aplicació per decidir quina part de la massa prostàtica correspon al grumoll i quina part es manté com a pròstata. La millora d'aquest algoritme de tall serà un dels objectius de l'altre part del macroprojecte. Per altra banda, a l'aplicació actual hi manquen diverses funcionalitats que són bàsiques en un sistema com el que ens ocupa. Entre aquestes funcionalitats hi trobem: cauteritzar tot tipus de ferides o teixits que puguin aparèixer durant l'operació i donar la possibilitat al cirurgià d'injectar aigua per tal de netejar la uretra en qualsevol moment. A més a més, cal afegir altres funcionalitats de control d'usuaris i arxius per tal de poder enregistrar els cirurgians i gestionar els diferents models prostàtics. Un altre aspecte que cal modificar és la manera d'interactuar amb el sistema. Fins ara la interacció es realitza via teclat i es vol que es faci mitjançant un *haptic* específic per a simulacions i amb diversos graus de llibertat. El nom de l'aplicació que es dissenya en aquest projecte s'anomena **Simpros**.

1.3 Gestió de riscos

Com tot projecte, aquest també pot tenir riscos esperats o inesperats. Per tal de poder prevenir-los de la millor manera, és necessari gestionar-los i tenir possibles solucions en cas de que apareguin. A continuació detallarem la previsió de riscos, les possibilitats per a mitigar-los i les probabilitats de que apareguin aquests riscos en el projecte:

Risc	Pla de contingència	Probabilitat
Impossibilitat de completar el projecte en el termini previst	- Augmentar la càrrega de treball diari. - Modificar la planificació i els objectius establerts inicialment.	Mitja
Manca de comunicació entre alguna de les parts del projecte	- Mantenir comunicació via e-mail i reunions setmanals amb el tutor del	Baixa

	projecte.	
Incompliment de la normativa establerta per la FIB	- Llegir la normativa específica per al projecte final de carrera i complir amb els terminis de totes les premisses.	Baixa

2 Anàlisi: Especificació

L'especificació és una branca de l'anàlisi que consisteix en la tasca de realitzar una descripció detallada d'un software. En aquest projecte s'ha decidit seguir la metodologia *Rational Unified Process (RUP)* per a la realització de l'especificació i el disseny del sistema. Aquesta metodologia s'utilitza en molts projectes i els resultats obtinguts són bastant òptims. En concret, tota la documentació i els diagrames obtinguts mitjançant l'aplicació d'aquesta metodologia, es troben en el cd de l'aplicació. L'especificació d'aquest projecte consta de descripcions dels requisits no funcionals, diagrames i descripcions dels casos d'ús, diagrama de classes, diagrames de seqüència del sistema, contractes de les operacions aparegudes en els diagrames de seqüència i realització d'altres diagrames interessants com són el d'estats o el de blocs en alt nivell.

2.1 Requeriments no funcionals

Recordem que els requeriments no funcionals han d'establir restriccions en el producte que està essent desenvolupat, en el procés de desenvolupament i en restriccions específiques que el producte pugui tenir. Els requeriments no funcionals de l'aplicació es mostren separats per branques i són els següents:

Interfície d'usuari i factors humans.

- El perfil de l'usuari del nostre sistema serà un usuari amb coneixements bàsics d'informàtica. Ha de conèixer el S.O. Linux i ha de saber moure's en ell.
- El sistema disposa d'una interfície gràfica simple, senzilla, intuïtiva i agradable, que és capaç de ser utilitzada fàcilment per a tot tipus d'usuaris no especialitzats.
- La interacció amb la aplicació es farà mitjançant un *haptic* de 6 graus de llibertat, amb ratolí i amb teclat.

Documentació

- El projecte ve acompanyat d'una sèrie de documents que formen part de la metodologia *Rational Unified Process (RUP)* en els que s'expliquen els detalls del sistema.

- L'aplicació conté menús d'ajuda en els que els usuaris podran consultar els seus dubtes en el ús de l'eina.

Consideracions de hardware

- Per al bon funcionament de l'aplicació s'utilitzarà com a mínim processadors Pentium III.
- La tarjeta gràfica haurà de ser de gama mitja/alta.

Tractament d'errors i condicions extremes

- En l'instant en el que es produeixi un error en el moviment que vulgui fer el cirurgià, el sistema mostrarà un avís d'error per pantalla per tal de que el cirurgià el detecti i pugui corregir-lo.

Factors de qualitat

- En cas de fallida o de problema extrem, el sistema es recuperarà amb facilitat i rapidesa.
- La portabilitat de l'aplicació a altres àmbits i llocs serà factible sempre i quan es disposi del hardware i el sistema operatiu necessari per executar l'aplicació en tota plenitud.

Modificacions al sistema

- El sistema està dissenyat per no impossibilitar modificacions, per tant té un disseny que permet i facilita possibles ampliacions i millores.

Temes de seguretat

- En el tema de la seguretat, els models de cadascun dels cirurgians estaran emmagatzemats en parts del disc dur concretes per cadascun dels cirurgians i requeriran d'identificació per poder-hi accedir ja que el sistema contindrà una sessió d'usuari per a cadascun dels usuaris actius.

Entorn de desenvolupament

- El llenguatge usat per desenvolupar l'aplicació haurà de ser C++ acompanyat de OpenGL per la part gràfica. A més, s'ha fet servir la tecnologia Qt per a la implementació de la capa de presentació.

- Caldrà disposar de les llibreries pertinents per a la part gràfica del sistema.

2.2 Diagrames de casos d'ús

En aquest apartat mostrem els diagrames de casos d'ús per a cadascun dels paquets de treball que conformen el sistema: Interaccions amb model i Gestió d'usuaris i fitxers. A més, també incorporem una descripció il·lustrativa dels possibles actors que intervenen en el nostre sistema.

2.2.1 Actors

Tal i com s'il·lustra en la Figura 2.1, en el nostre sistema trobem un usuari genèric i dos possibles usuaris concrets que són l'administrador del sistema i el cirurgià.

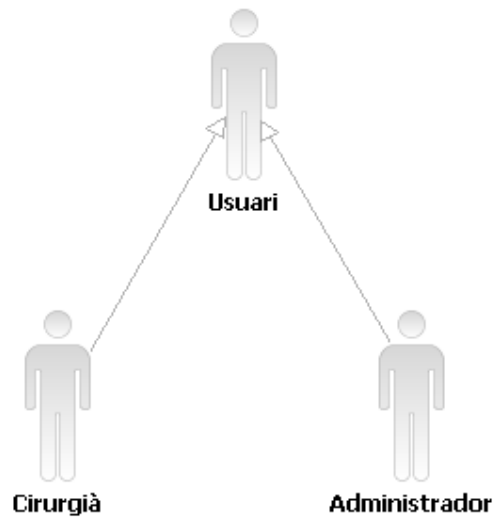


Figura 2.1: Actors del sistema

2.2.2 Interaccions amb model

La Figura 2.2 il·lustra el diagrama de casos d'ús pel paquet de treball Interaccions amb model. Aquests casos d'ús són els casos d'ús purs de la simulació amb el model prostàtic. Per tant, l'únic usuari que apareix relacionat amb els casos d'ús d'aquest paquet de treball és el cirurgià. Hem de remarcar que el cas d'ús *Tallar* requereix que prèviament s'hagi realitzat el cas d'ús *Activar pas de corrent nivell de tall* i que el cas d'ús *Cauteritzar* requereix que prèviament s'hagi realitzat el cas d'ús *Activar pas de corrent nivell de Cauterització*. Això és així ja que els cirurgians disposen de dos nivells de corrent diferents per a cadascuna d'aquestes dues funcionalitats.

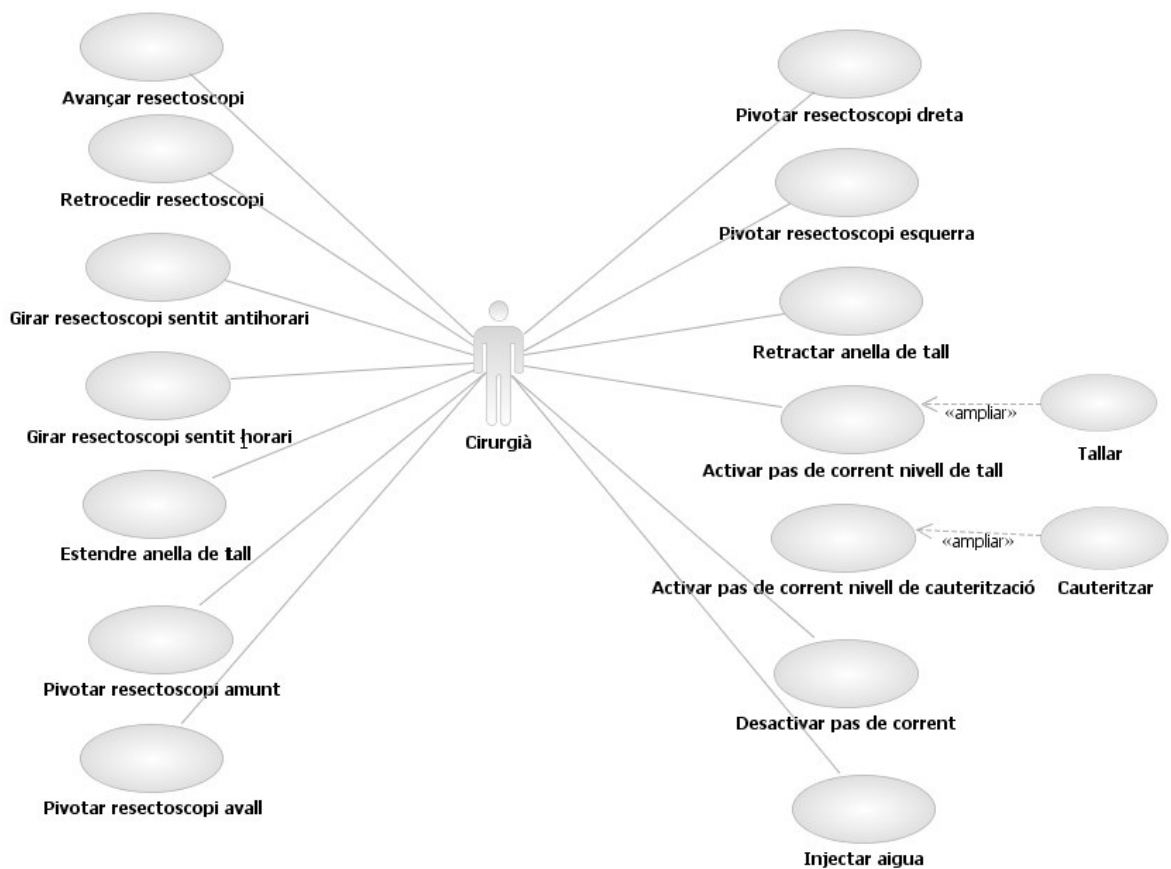


Figura 2.2: Diagrama de casos d'ús: Interaccions amb model

2.2.3 Gestió d'usuaris i fitxers

La Figura 2.3 il·lustra el diagrama de casos d'ús pel paquet de treball Gestió d'usuaris i fitxers. Aquests casos d'ús serveixen per a gestionar tota la branca dels usuaris del sistema i la branca dels fitxers que corresponen a models prostàtics. Per tant, els usuaris relacionats amb els casos d'ús d'aquest paquet de treball són tant l'administrador com el cirurgià.

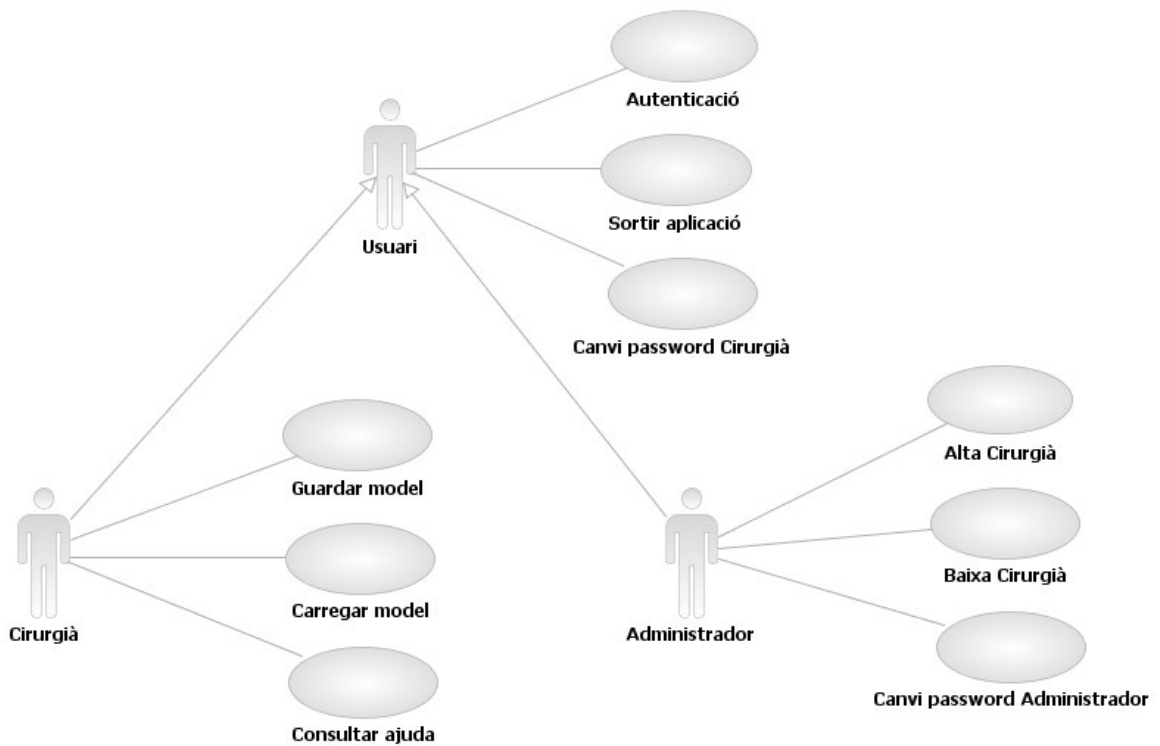


Figura 2.3: Diagrama de casos d'ús: Gestió d'usuaris i fitxers

2.3 Descripcions dels casos d'ús

En aquest apartat realitzem la descripció dels casos d'ús del sistema, que serveixen per conèixer les interaccions entre els diferents actors de cadascun d'aquests casos d'ús. Els components d'aquesta descripció són els següents:

- **Identificador:** identificador únic i representatiu del cas d'ús. La codificació d'aquest identificador ve donada per la simbologia **XYX** on X representa el paquet de treball i YY són els identificadors del propi cas d'ús.
- **Descripció:** descripció informal del propòsit del cas d'ús.
- **Actors:** llista d'actors que intervenen en el cas d'ús.
- **Flux bàsic:** descripció detallada de la interacció entre actors i sistema.
- **Flux alternatiu:** descripció de les excepcions al flux bàsic.

2.3.1 Interaccions amb model

Tot seguit descrivim en detall cadascun dels casos d'ús del paquet de treball Interaccions amb model.

Avançar resectoscopi

- **Identificador:** UC-101
- **Descripció:** Aquest cas d'us permet al Cirurgià desplaçar el resectoscopi cap a dins de la uretra.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol avançar arbitràriament el resectoscopi.	2. El sistema comprova que el moviment és factible i el realitza.

Flux Alternatiu:

2. El moviment no és factible, es desencadena el tractament de l'excepció.

Retrocedir resectoscopi

- **Identificador:** UC-102
- **Descripció:** Aquest cas d'us permet al Cirurgià desplaçar el resectoscopi cap a fora de la uretra.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol retrocedir arbitràriament el resectoscopi.	2. El sistema comprova que el moviment és factible i el realitza.

Flux Alternatiu:

2. El moviment no és factible, es desencadena el tractament de l'excepció.

Girar resectoscopi sentit antihorari

- **Identificador:** UC-103
- **Descripció:** Aquest cas d'us permet al Cirurgià girar el resectoscopi en sentit antihorari.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol girar arbitràriament el resectoscopi en sentit antihorari.	2. El sistema comprova que el gir és factible i el realitza.

Flux Alternatiu:

2. El gir no és factible, es desencadena el tractament de l'excepció.

Girar resectoscopi sentit horari

- **Identificador:** UC-104
- **Descripció:** Aquest cas d'us permet al Cirurgià girar el resectoscopi en sentit horari.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol girar arbitràriament el resectoscopi en sentit horari.	2. El sistema comprova que el gir és factible i el realitza.

Flux Alternatiu:

2. El gir no és factible, es desencadena el tractament de l'excepció.

Estendre anella de tall

- **Identificador:** UC-105
- **Descripció:** Aquest cas d'us permet al Cirurgià estendre l'anella de tall respecte de la cànula del resectoscopi.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol estendre l'anella de tall arbitràriament.	2. El sistema comprova que el moviment és factible i el realitza.

Flux Alternatiu:

2. El moviment no és factible, es desencadena el tractament de l'excepció.

Retractar anella de tall

- **Identificador:** UC-106
- **Descripció:** Aquest cas d'us permet al Cirurgià retractar l'anella de tall respecte de la cànula del resectoscopi.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol retractar l'anella de tall arbitràriament.	2. El sistema comprova que el moviment és factible i el realitza.

Flux Alternatiu:

2. El moviment no és factible, es desencadena el tractament de l'excepció.

Pivotar resectoscopi amunt

- **Identificador:** UC-107
- **Descripció:** Aquest cas d'us permet al Cirurgià pivotar el resectoscopi respecte del seu punt de recolzament cap amunt.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol pivotar el resectoscopi respecte del seu punt de recolzament cap amunt arbitràriament.	2. El sistema comprova que el moviment és factible i el realitza.

Flux Alternatiu:

2. El moviment no és factible, es desencadena el tractament de l'excepció.

Pivotar resectoscopi avall

- **Identificador:** UC-108
- **Descripció:** Aquest cas d'us permet al Cirurgià pivotar el resectoscopi respecte del seu punt de recolzament cap avall.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol pivotar el resectoscopi respecte del seu punt de recolzament cap avall arbitràriament.	2. El sistema comprova que el moviment és factible i el realitza.

Flux Alternatiu:

2. El moviment no és factible, es desencadena el tractament de l'excepció.

Pivotar resectoscopi dreta

- **Identificador:** UC-109
- **Descripció:** Aquest cas d'us permet al Cirurgià pivotar el resectoscopi respecte del seu punt de recolzament cap a la dreta.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol pivotar el resectoscopi respecte del seu punt de recolzament cap a la dreta arbitràriament.	2. El sistema comprova que el moviment és factible i el realitza.

Flux Alternatiu:

2. El moviment no és factible, es desencadena el tractament de l'excepció.

Pivotar resectoscopi esquerra

- **Identificador:** UC-110
- **Descripció:** Aquest cas d'us permet al Cirurgià pivotar el resectoscopi respecte del seu punt de recolzament cap a l'esquerra.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol pivotar el resectoscopi respecte del seu punt de recolzament cap a l'esquerra arbitràriament.	2. El sistema comprova que el moviment és factible i el realitza.

Flux Alternatiu:

2. El moviment no és factible, es desencadena el tractament de l'excepció.

Activar pas de corrent nivell de tall

- **Identificador:** UC-111
- **Descripció:** Aquest cas d'us permet al Cirurgià activar el pas de corrent per l'anella de tall per tal d'efectuar el procés de tall de material.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol activar el pas de corrent de l'anella de tall amb nivell	2. El sistema activa el pas de corrent a través de l'anella.

de tall.	
----------	--

Activar pas de corrent nivell de cauterització

- **Identificador:** UC-112
- **Descripció:** Aquest cas d'us permet al Cirurgià activar el pas de corrent per l'anella de tall per tal d'efectuar la cauterització d'una ferida.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol activar el pas de corrent de l'anella de tall amb nivell de cauterització.	2. El sistema activa el pas de corrent a través de l'anella.

Desactivar pas de corrent

- **Identificador:** UC-113
- **Descripció:** Aquest cas d'us permet al Cirurgià desactivar el pas de corrent per l'anella de tall.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol desactivar el pas de corrent de l'anella de tall.	2. El sistema desactiva el pas de corrent a través de l'anella.

Tallar

- **Identificador:** UC-114
- **Descripció:** Aquest cas d'us permet al Cirurgià tallar material dins de la uretra.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol realitzar l'operació de tall de material mitjançant un moviment arbitrari que provoca inici de col·lisió amb la pròstata i amb una quantitat de força no nul·la.	2. El sistema comprova que les condicions de tall siguin les adequades i mostra la nova posició del resectoscopi si existeix contacte amb la pròstata o mostra com el tros tallat es perd per dins de la uretra si deixa d'existir contacte.

Flux Alternatiu:

2. El pas de corrent de l'anella de tall amb nivell de tall no està activat, es desencadena el tractament de l'excepció.

Cauteritzar

- **Identificador:** UC-115
- **Descripció:** Aquest cas d'us permet al Cirurgià cauteritzar qualsevol ferida produïda durant l'operació.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol cauteritzar la posició actual si es produeix	2. El sistema comprova que el pas de corrent per l'anella de tall estigui activat i mostra

una situació de contacte superficial amb la pròstata amb una quantitat de força nul·la.	l'estat cicatritzat de la zona cauteritzada.
---	--

Flux Alternatiu:

2. El pas de corrent per l'anella de tall amb nivell de cauterització no està activat, es desencadena el tractament de l'excepció.

Injectar aigua

- **Identificador:** UC-116
- **Descripció:** Aquest cas d'us permet al Cirurgià injectar aigua durant l'operació per tal de netejar les diferents zones.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol injectar aigua en la posició denotada.	2. El sistema actualitza l'aspecte de la uretra un cop netejada la zona.

2.3.2 Gestió d'usuaris i fitxers

Tot seguit descrivim en detall cadascun dels casos d'ús del paquet de treball Gestió d'usuaris i fitxers.

Autenticació

- **Identificador:** UC-201
- **Descripció:** Aquest cas d'us permet a l'Usuari autenticar-se i carregar la seva pròpia sessió.
- **Actors:**
 - Usuari: Usuari genèric de l'aplicació.

Flux Bàsic:

Actors	Sistema
1. L'actor Usuari indica al sistema que vol autenticar-se passant-li el nom d'usuari i la contrasenya.	2. El sistema comprova que les dades proporcionades per l'actor Usuari siguin correctes i carrega la seva pròpia sessió.

Flux Alternatiu:

2. No existeix un Usuari amb el nom d'usuari i la contrasenya proporcionades per l'actor Usuari, es desencadena el tractament de l'excepció i torna a 1.

Sortir aplicació

- **Identificador:** UC-202
- **Descripció:** Aquest cas d'us permet a l'Usuari sortir de l'aplicació un cop hagi finalitzat la seva simulació.
- **Actors:**
 - Usuari: Usuari genèric de l'aplicació.

Flux Bàsic:

Actors	Sistema
1. L'actor Usuari indica al sistema que vol sortir de l'aplicació.	2. Si el model actual no ha estat guardat prèviament llavors el sistema dona la

	possibilitat de guardar-lo a l'actor Usuari. Si no llavors el sistema tanca l'aplicació.
3. L'actor Usuari confirma l'acció de guardar el model actual.	4. El sistema guarda el model actual i tanca l'aplicació.

Canvi password Cirurgia

- **Identificador:** UC-203
- **Descripció:** Aquest cas d'us permet al Cirurgià canviar la seva contrasenya personal o a l'administrador canviar la contrasenya d'un Cirurgià qualsevol.
- **Actors:**
 - Usuari: Usuari genèric de l'aplicació.

Flux Bàsic:

Actors	Sistema
1. Si l'actor Usuari és un Cirurgià llavors indica al sistema que vol canviar la seva contrasenya tot passant-li la contrasenya actual i la nova. Si per contra, l'actor Usuari és l'Administrador llavors inclou el nom d'usuari del Cirurgià corresponent i la nova contraseya.	2. El sistema demana confirmació a l'actor Usuari.
3. L'actor Usuari confirma l'acció tornant a proporcionar la nova contrasenya.	4. El sistema comprova que les dues contrasenyes proporcionades per l'actor Usuari són les mateixes i realitza el canvi de contrasenya.

Flux Alternatiu:

2. La contrasenya actual de l'actor Cirurgià no és correcte, es desencadena el tractament de l'excepció i torna a 1.

4. Les contrasenyes proporcionades per l'actor Usuari no coincideixen, es desencadena el tractament de l'excepció i torna a 3.

Guardar model

- **Identificador:** UC-204
- **Descripció:** Aquest cas d'us permet al Cirurgià guardar el model actual de la simulació per poder carregar-ho en un altre moment.
- **Actors:**
 - Cirurgià: Cirurgià que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgià indica al sistema que vol guardar el model actual.	2. El sistema demana el directori destí i el nom del fitxer al Cirurgià.
3. El Cirurgià indica el directori destí i el nom del fitxer corresponents.	4. El sistema comprova que les dades proporcionades pel Cirurgià són correctes. Si ja existeix un fitxer amb el nom proporcionat pel Cirurgià al directori proporcionat pel Cirurgià llavors el sistema dona la possibilitat de reemplaçar l'antic fitxer per l'actual.

Flux Alternatiu:

4. El directori destí no existeix, es desencadena el tractament de l'excepció.

Carregar model

- **Identificador:** UC-205
- **Descripció:** Aquest cas d'us permet al Cirurgià carregar un model d'una sessió prèviament guardada.
- **Actors:**

- Cirurgia: Cirurgia que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgia indica al sistema que vol carregar un model.	2. El sistema demana el directori origen del model a carregar al Cirurgia.
3. El Cirurgia indica el directori origen del model.	4. El sistema comprova que existeixin models en el directori proporcionat pel Cirurgia i llista els models corresponents.
5. El Cirurgia indica el model que vol carregar.	6. El sistema carrega el model proporcionat pel Cirurgia i el mostra.

Flux Alternatiu:

- 4. El directori origen no existeix, es desencadena el tractament de l'excepció.
- 4. No existeixen models en el directori origen proporcionat pel Cirurgia, es desencadena el tractament de l'excepció i torna a 3.

Consultar ajuda

- **Identificador:** UC-206
- **Descripció:** Aquest cas d'us permet al Cirurgia consultar l'ajuda de l'aplicació.
- **Actors:**
 - Cirurgia: Cirurgia que es troba en període d'aprenentatge pel que fa a les operacions de pròstata.

Flux Bàsic:

Actors	Sistema
1. El Cirurgia indica al sistema que vol consultar l'ajuda.	2. El sistema mostra l'ajuda de l'aplicació.

Alta Cirurgia

- **Identificador:** UC-207

- **Descripció:** Aquest cas d'us permet a l'Administrador donar d'alta a un nou usuari Cirurgià.
- **Actors:**
 - Administrador: Administrador que s'encarrega de fer la gestió d'usuaris.

Flux Bàsic:

Actors	Sistema
1. L'actor Administrador diu que vol donar d'alta a un nou Cirurgià tot passant-li el nom d'usuari i la contrasenya.	2. El sistema dona d'alta a un nou Cirurgià amb les dades proporcionades per l'administrador.

Flux Alternatiu:

2. Ja existeix un Cirurgià amb aquest nom d'usuari, es desencadena el tractament de l'excepció i torna a 1.

Baixa Cirurgià

- **Identificador:** UC-208
- **Descripció:** Aquest cas d'us permet a l'Administrador donar de baixa a un usuari Cirurgià.
- **Actors:**
 - Administrador: Administrador que s'encarrega de fer la gestió d'usuaris.

Flux Bàsic:

Actors	Sistema
1. L'actor Administrador diu que vol donar de baixa a un Cirurgià tot passant-li el nom d'usuari.	2. El sistema dona de baixa al Cirurgià amb les dades proporcionades per l'administrador.

Flux Alternatiu:

2. No existeix un Cirurgià amb aquest nom d'usuari, es desencadena el tractament de l'excepció i torna a 1.

Canvi password Administrador

- **Identificador:** UC-209
- **Descripció:** Aquest cas d'us permet a l'Administrador canviar la seva contrasenya personal.
- **Actors:**
 - Administrador: Administrador que s'encarrega de fer la gestió d'usuaris.

Flux Bàsic:

Actors	Sistema
1. L'actor Administrador indica al sistema que vol canviar la seva contrasenya tot passant-li la contrasenya actual i la nova.	2. El sistema demana confirmació a l'Administrador.
3. L'actor Administrador confirma l'acció tornant a proporcionar la nova contrasenya.	4. El sistema comprova que les dues contrasenyes proporcionades per l'Administrador són les mateixes i realitza el canvi de contrasenya.

Flux Alternatiu:

2. La contrasenya actual no és correcte, es desencadena el tractament de l'excepció i torna a 1.
4. Les contrasenyes proporcionades per l'Administrador no coincideixen, es desencadena el tractament de l'excepció i torna a 3.

2.4 Diagrames de blocs i d'estat

En aquest apartat mostrem per una banda un diagrama de molt alt nivell que especifica els blocs bàsics de l'aplicació i per l'altra banda, els diagrames d'estat que il·lustren els diferents estats en els que pot estar l'aplicació tenint en compte les diverses funcionalitats de la part d'interacció amb el model tant a baix nivell com a alt nivell. La Figura 2.4 il·lustra els blocs bàsics de l'aplicació en alt nivell.

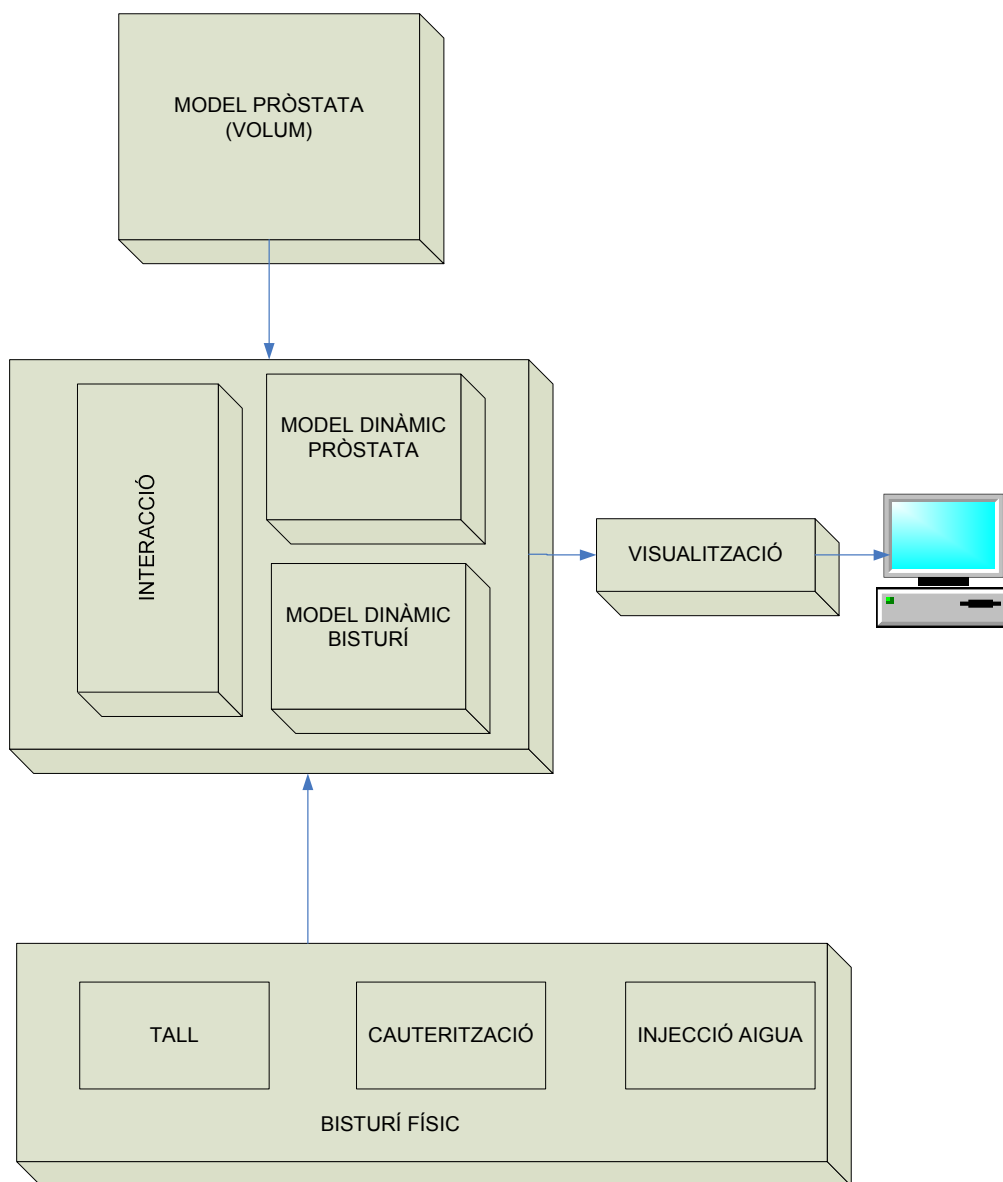


Figura 2.4: Diagrama de blocs

En la Figura 2.5 observem que l'interactuador es queda en un bucle esperant a rebre events per part del cirurgià un cop s'ha carregat un model prostàtic. La manera de que l'interactuador finalitzi aquest procés és sortint de l'aplicació. Això simplifica el funcionament a alt nivell de l'aplicació.

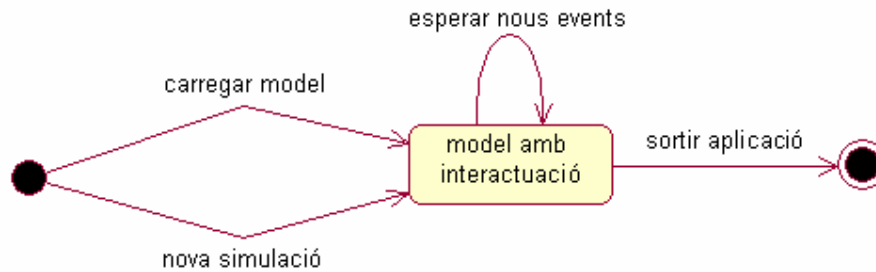


Figura 2.5: Diagrama d'estats a alt nivell

En la Figura 2.6 observem que hi ha sis possibles estats a part de l'estat inicial. Aquests estats depenen de tres variables fonamentals en el nostre sistema: les col·lisions entre el resectoscopi i la massa prostàtica, el corrent de tipus tall i el corrent de tipus cauterització. En la Figura 2.6 la paraula "moure" equival a qualsevol tipus de moviment possible que realitzi el cirurgià durant la simulació.

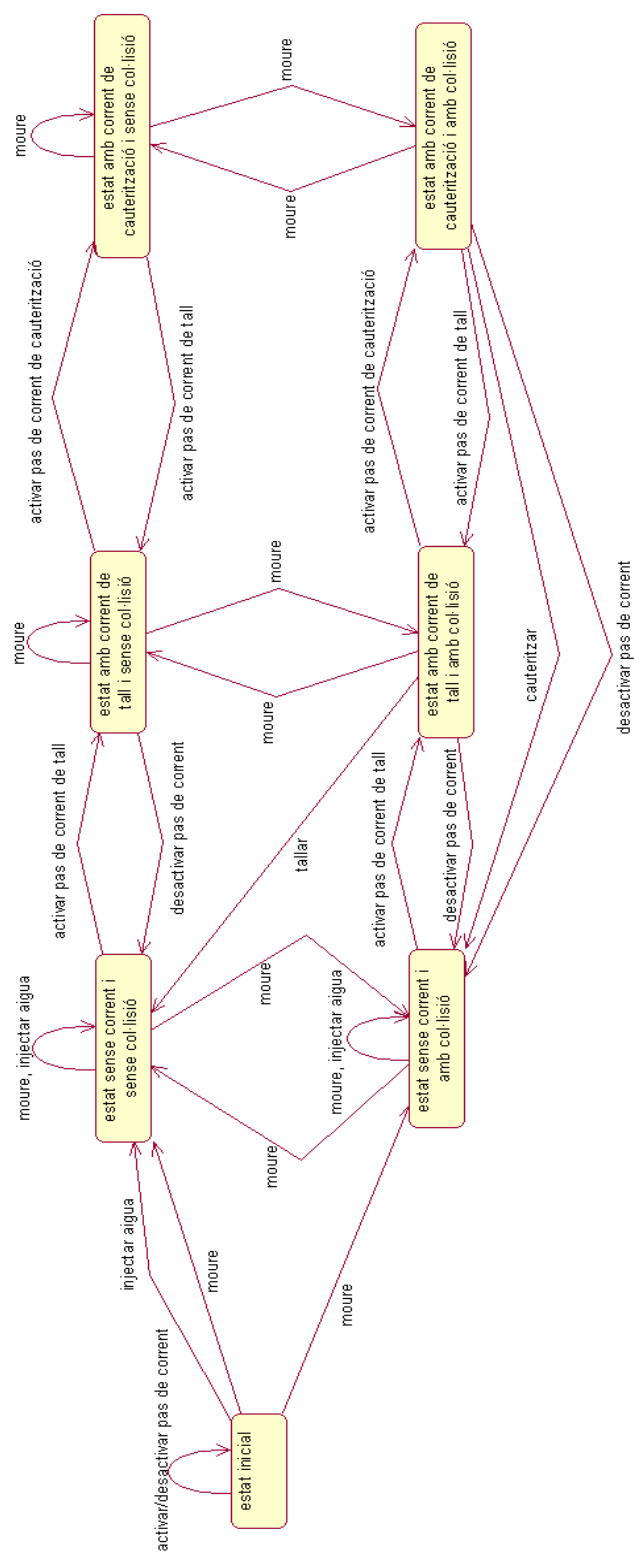


Figura 2.6: Diagrama d'estats a baix nivell

2.5 Diagrames de classes

Inicialment, hem de definir dues tuples que apareixeran al llarg dels diagrames de classes posteriors. La primera d'aquestes és Punt3D i conté tres atributs que corresponen a les components dels eixos cardinals. La segona és Component i conté les components dels eixos cardinals, els angles de desplaçament respecte el punt de recolzament dels eixos x i y; i finalment, l'angle respecte l'eix z que és l'encarregat de mesurar els girs horaris/antihoraris. La Figura 2.7 il·lustra l'especificació de les tuples Punt3D i Component:



Figura 2.7: Especificació de Punt3D i Component

Un cop ja coneixem el significat d'aquestes dues tuples creades específicament per a la nostra especificació, podem fer ús d'elles en els models conceptuals. Primer de tot, especificarem el model conceptual pel paquet de treball Interaccions amb model, que és l'essència de l'aplicació. La classe Model representa al **model prostàtic**, el qual té dues branques clarament definides que són el **resectoscopi** i la **massa prostàtica**. Aquest model està identificat pel seu nom i pel *path* on estigui ubicat si és que ha estat guardat prèviament. La classe Interactuador és l'encarregada de rebre i processar els events generats pel cirurgià i la classe Visualitzador visualitza l'escena amb tots els seus components. Es genera una classe associativa Impacte en el moment en el que apareix col·lisió entre les dues branques principals del model: el resectoscopi i la massa prostàtica. Per una banda, el resectoscopi s'identifica per la seva posició dins l'escena (com a posició agafarem la posició que estigui més a prop de la bufeta) i, per altra banda, la massa prostàtica contindrà tres identificadors que corresponen a cadascun dels números de vòxels (unitat utilitzada en l'aplicació que correspon a un píxel tridimensional) dels tres eixos cardinals on trobem massa prostàtica. La Figura 2.8 mostra l'especificació del paquet de treball Interaccions amb model:

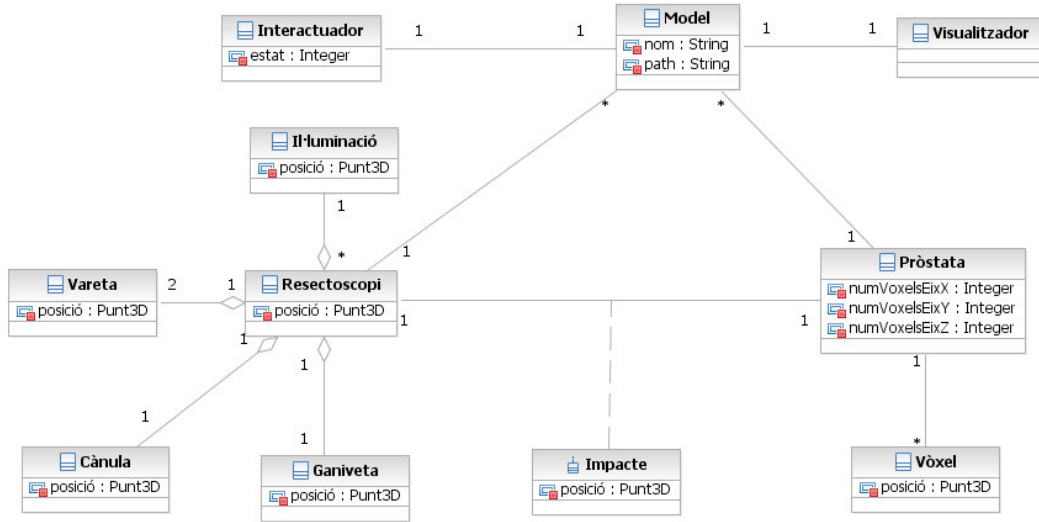


Figura 2.8: Diagrama de classes: Interaccions amb model

A continuació especificuem el diagrama de classes pel paquet de treball Gestió d'usuaris i fitxers. Aquest diagrama il·lustra els dos tipus d'usuaris que existeixen i la relació entre l'usuari cirurgià i els models prostàtics dels que disposa. La Figura 2.9 mostra l'especificació del paquet de treball Gestió d'usuaris i fitxers:

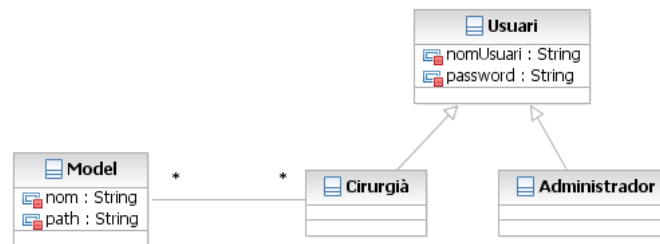


Figura 2.9: Diagrama de classes: Gestió d'usuaris i fitxers

2.6 Diagrames de seqüència del sistema

En aquest apartat mostrem els diagrames de seqüència de l'especificació de l'aplicació, en els quals apareixen les operacions entre els actors i el propi sistema. Aquests diagrames de seqüència formen part del model del comportament del sistema juntament amb els contractes de les operacions que es defineixen en el següent apartat.

2.6.1 Interaccions amb model

Hi ha un conjunt de casos d'ús que utilitzen una única operació anomenada Moviment, que és l'encarregada de verificar els moviments realitzats pel cirurgià amb el resectoscopi i també els talls corresponents. Els casos d'ús referenciats prèviament són: Avançar resectoscopi, Retrocedir resectoscopi, Girar resectoscopi sentit antihorari, Girar resectoscopi sentit horari, Pivotar resectoscopi amunt, Pivotar resectoscopi avall, Pivotar resectoscopi dreta, Pivotar resectoscopi esquerra i Tallar. La Figura 2.10 il·lustra el diagrama de seqüència d'aquesta operació que comunica al cirurgià amb el sistema.

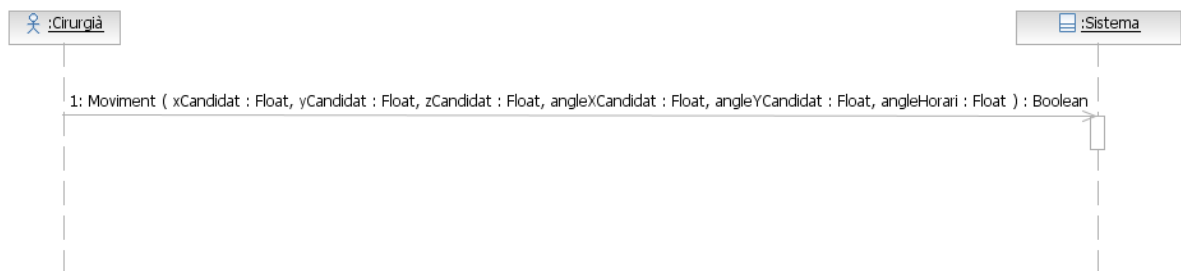


Figura 2.10: Diagrama de seqüència de l'operació Moviment

Hi ha un conjunt de casos d'ús que utilitzen una única operació anomenada TraslladarVaretes, que és l'encarregada de traslladar les varetes de l'anella de tall cap endavant o cap endarrere quan sigui necessari. Els casos d'ús referenciats prèviament són: Estendre anella de tall i Retractar anella de tall. La Figura 2.11 il·lustra el diagrama de seqüència d'aquesta operació que comunica al cirurgià amb el sistema.

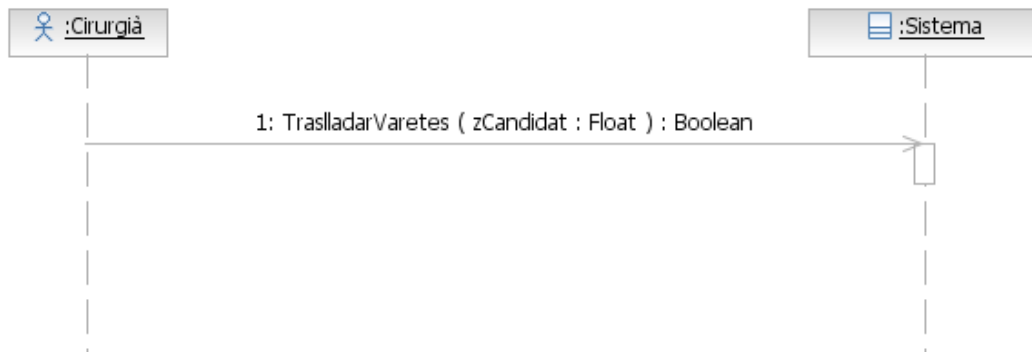


Figura 2.11: Diagrama de seqüència de l'operació TrasladarVarettes

Hi ha un conjunt de casos d'ús que utilitzen una única operació anomenada *AlterarPasDeCorrent*, que és l'encarregada d'alterar o desactivar el pas de corrent que circula per l'anella de tall. Aquest pas de corrent està especificat com un enter en el que el 0 és desactivat, el 1 és nivell de tall i el 2 és nivell de cauterització. És important afirmar que els nivells de tall i cauterització no són iguals ja que el de tall té més intensitat, per poder tallar de manera solvent. Els casos d'ús referenciats prèviament són: *Activar pas de corrent de nivell de tall*, *Activar pas de corrent de nivell de cauterització* i *Desactivar pas de corrent*. La Figura 2.12, la Figura 2.13 i la Figura 2.14 il·lustren els diagrames de seqüència d'aquesta operació que comunica al cirurgià amb el sistema pels casos d'ús *Activar pas de corrent de nivell de tall*, *Activar pas de corrent de nivell de cauterització* i *Desactivar pas de corrent* respectivament.

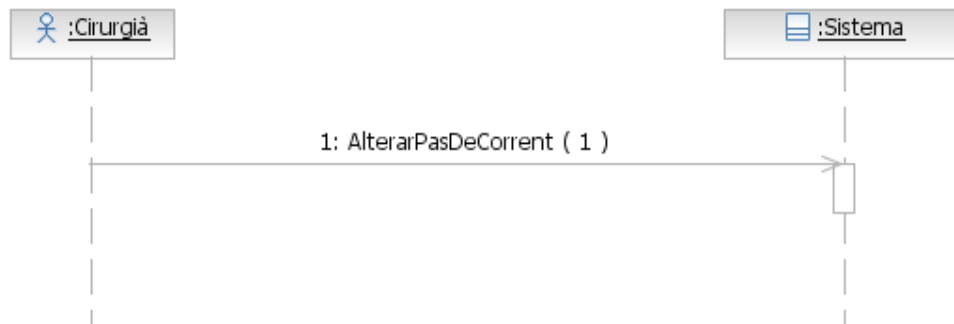


Figura 2.12: Diagrama de seqüència de l'operació AlterarPasDeCorrent; nivell de tall

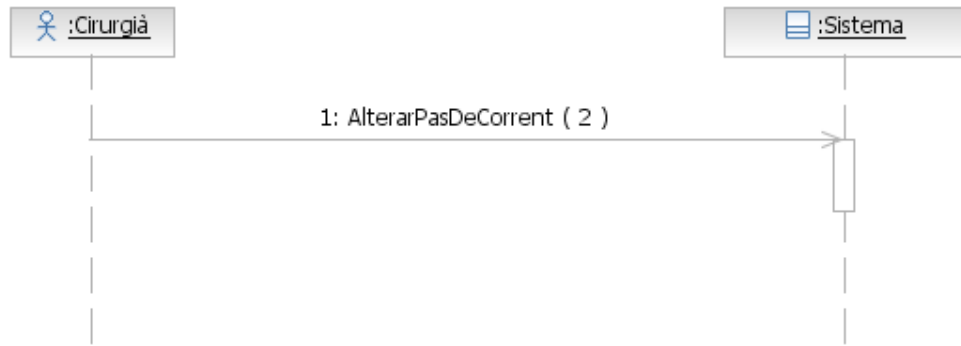


Figura 2.13: Diagrama de seqüència de l'operació AlterarPasDeCorrent; nivell de cauterització



Figura 2.14: Diagrama de seqüència de l'operació AlterarPasDeCorrent; desactivat

Pel cas d'ús Cauteritzar, especifiquem l'operació Cauteritzar. Aquesta operació és l'encarregada de cauteritzar una zona determinada de la pròstata. La Figura 2.15 il·lustra el diagrama de seqüència d'aquesta operació que comunica el cirurgià amb el sistema.

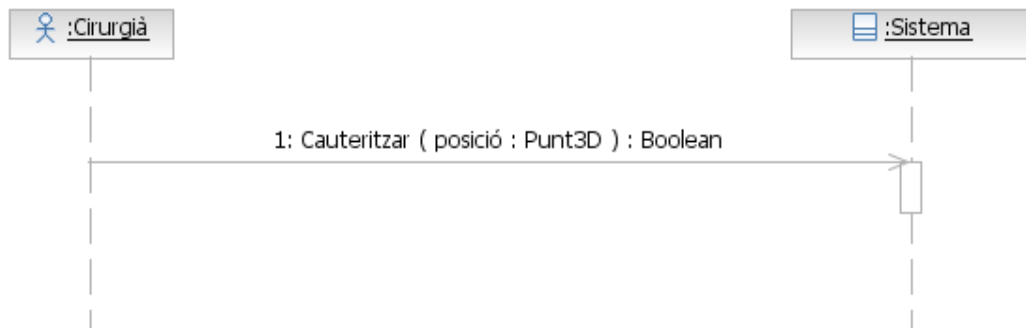


Figura 2.15: Diagrama de seqüència de l'operació Cauteritzar

Pel cas d'ús Injectar Aigua, especifiquem l'operació InjectarAigua. Aquesta operació és l'encarregada d'injectar aigua en una zona determinada per a netejar-la. La Figura 2.16 il·lustra el diagrama de seqüència d'aquesta operació que comunica el cirurgià amb el sistema.

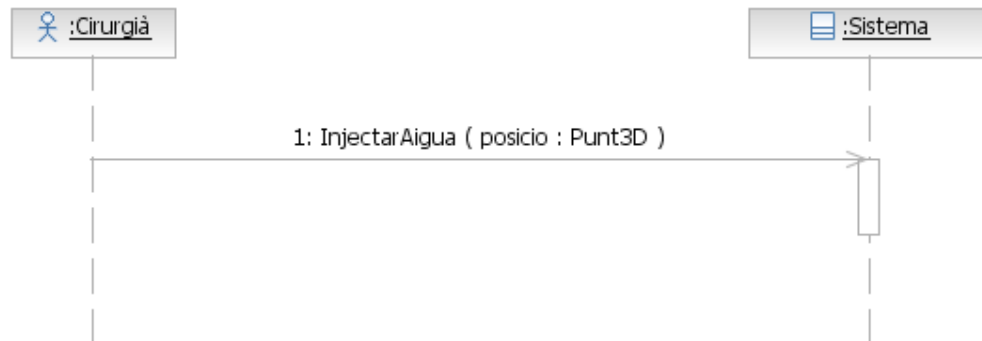


Figura 2.16: Diagrama de seqüència de l'operació InjectarAigua

2.6.2 Gestió d'usuaris i fitxers

Pel cas d'ús Autenticació, especifiquem l'operació Autenticar. Aquesta operació és l'encarregada d'autenticar un usuari en l'aplicació. La Figura 2.17 il·lustra el diagrama de seqüència d'aquesta operació que comunica a un usuari amb el sistema.

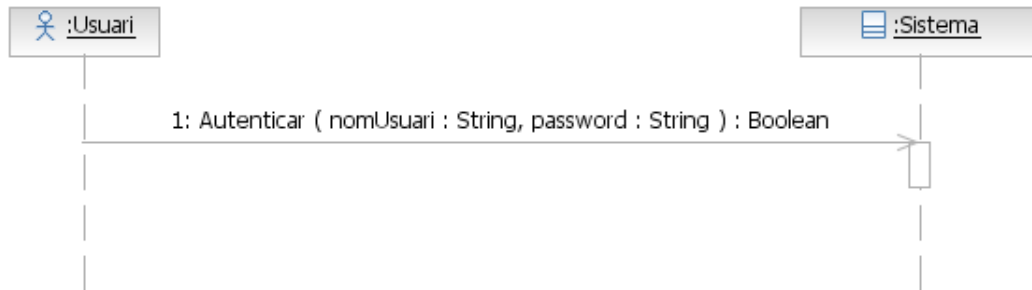


Figura 2.17: Diagrama de seqüència de l'operació Autenticar

Pel cas d'ús Sortir Aplicació, especifiquem les operacions GuardarModel i SortirAplicació. La primera operació fa possible guardar un model abans de sortir de l'aplicació si el model no ha estat guardat prèviament. La segona operació és l'encarregada de sortir de l'aplicació. La Figura 2.18 il·lustra el diagrama de seqüència d'aquestes dues operacions que comuniquen a un usuari amb el sistema.

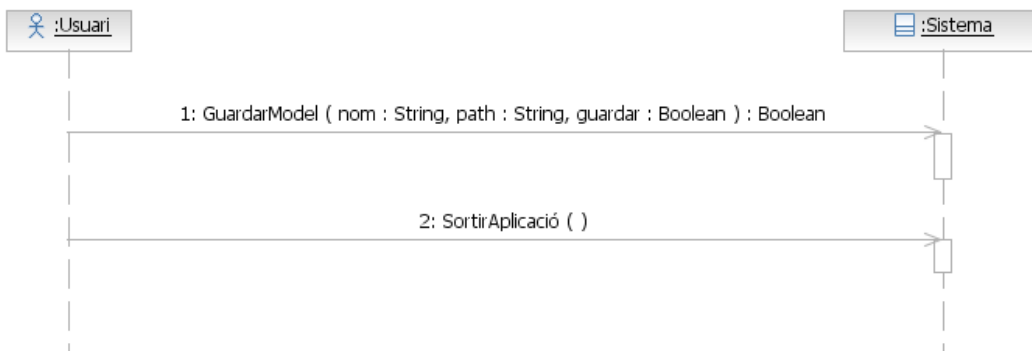


Figura 2.18: Diagrama de seqüència de les operacions GuardarModel i SortirAplicació

Hi ha un conjunt de casos d'ús que utilitzen una única operació anomenada CanviPassword, l'encarregada de canviar la contrasenya d'un usuari genèric. En el cas del canvi de contrasenya dels cirurgians, només s'inclouen els canvis dels propis cirurgians. Els casos d'ús referenciats prèviament són: Canvi password Cirurgià i Canvi password Administrador. La Figura 2.19 il·lustra el diagrama de seqüència d'aquesta operació que comunica a un usuari amb el sistema.

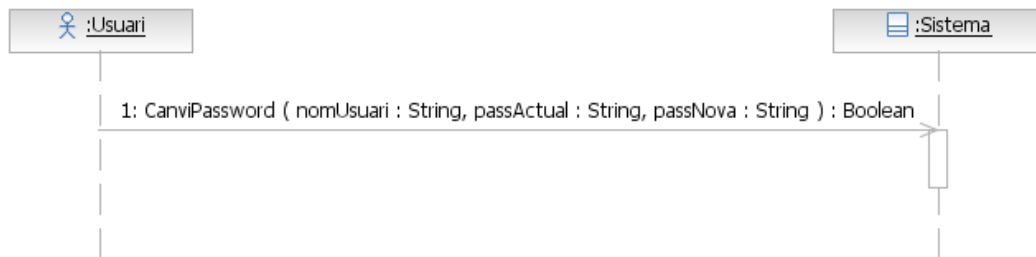


Figura 2.19: Diagrama de seqüència de l'operació CanviPassword

L'operació anomenada CanviPassCirViaAdm, és l'encarregada de canviar la contrasenya dels cirurgians en el cas en el que el canvi el realitzi l'administrador. En aquest cas, no és necessari conèixer la contrasenya actual del cirurgià. El cas d'ús que utilitza aquesta operació és Canvi password Cirurgià. La Figura 2.20 il·lustra el diagrama de seqüència d'aquesta operació que comunica a l'administrador amb el sistema.

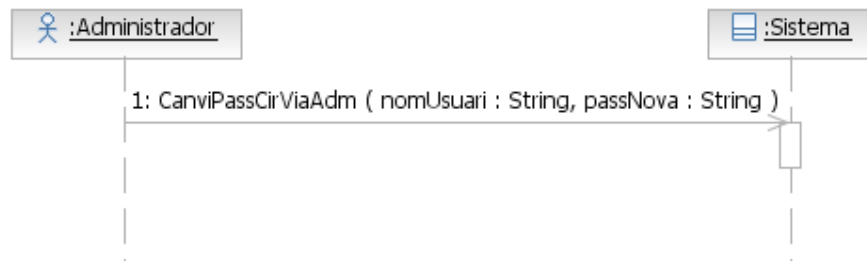


Figura 2.20: Diagrama de seqüència de l'operació CanviPassCirViaAdm

Pel cas d'ús Guardar Model, especifiquem l'operació GuardarModel. Aquesta operació serà l'encarregada de guardar l'estat del model prostàtic en un fitxer. La Figura 2.21 il·lustra el diagrama de seqüència d'aquesta operació que comunica al cirurgià amb el sistema.

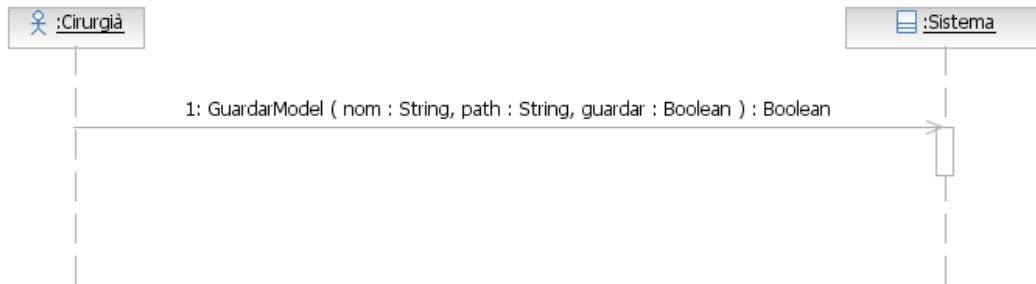


Figura 2.21: Diagrama de seqüència de l'operació GuardarModel

Pel cas d'ús Carregar Model, especifiquem l'operació CarregarModel. Aquesta operació serà l'encarregada de carregar l'estat d'un model prostàtic des d'un fitxer. La Figura 2.22 il·lustra el diagrama de seqüència d'aquesta operació que comunica al cirurgià amb el sistema.

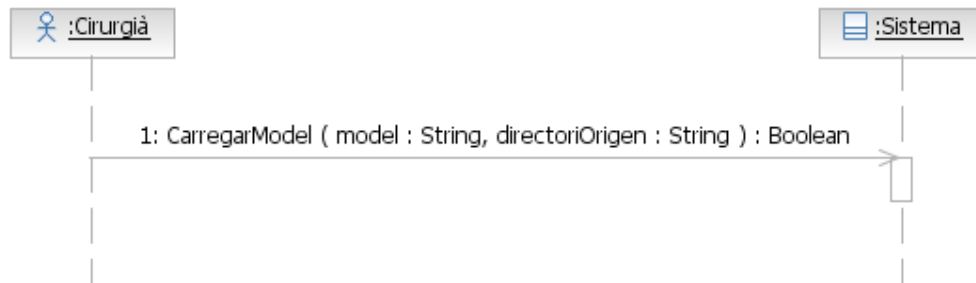


Figura 2.22: Diagrama de seqüència de l'operació CarregarModel

Pel cas d'ús Consultar Ajuda, especifiquem l'operació ConsultarAjuda. Aquesta operació serà l'encarregada de mostrar-li al cirurgià l'ajuda sobre el funcionament de l'aplicació. La Figura 2.23 il·lustra el diagrama de seqüència d'aquesta operació que comunica al cirurgià amb el sistema.

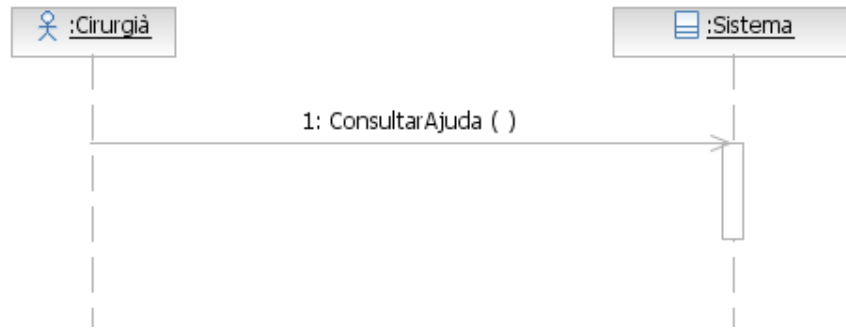


Figura 2.23: Diagrama de seqüència de l'operació ConsultarAjuda

Pel cas d'ús Alta Cirurgià, especifiquem l'operació AltaCirugià. Aquesta operació serà l'encarregada de donar d'alta a un cirurgià en el sistema. La Figura 2.24 il·lustra el diagrama de seqüència d'aquesta operació que comunica a l'administrador amb el sistema.

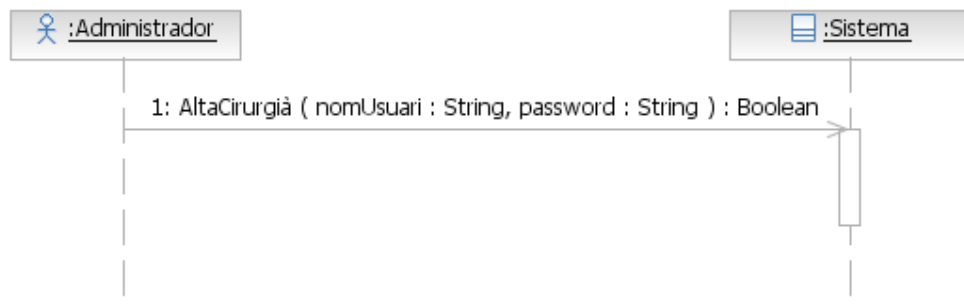


Figura 2.24: Diagrama de seqüència de l'operació AltaCirugià

Pel cas d'ús Baixa Cirugià, especifiquem l'operació BaixaCirugià. Aquesta operació serà l'encarregada de donar de baixa a un cirurgià en el sistema. La Figura 2.25 il·lustra el diagrama de seqüència d'aquesta operació que comunica a l'administrador amb el sistema.

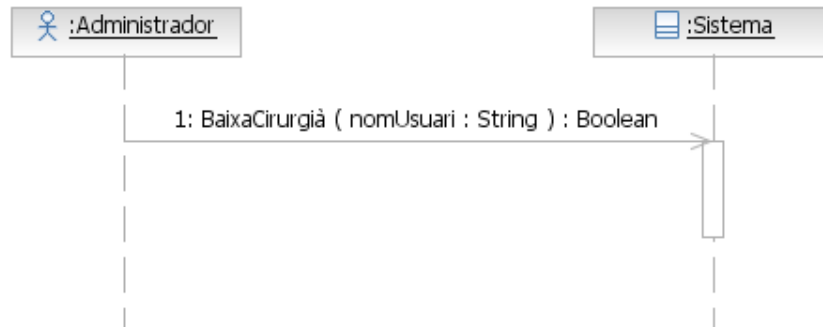


Figura 2.25: Diagrama de seqüència de l'operació BaixaCirurgia

2.7 Contractes de les operacions del sistema

Els contractes de les operacions descriuen el comportament del sistema per constatar quins són els canvis d'estat d'informació i quines són les sortides que el sistema proporciona quan s'invoca una operació.

El tipus de descripció que es proporciona és totalment declarativa donant més importància al que realment fa l'operació que no a com ho fa. Els components dels contractes de les operacions són els següents:

- **Nom:** indica el nom i els atributs de l'operació.
- **Responsabilitats:** descripció informal del propòsit de l'operació.
- **Excepcions:** descripció de la reacció del sistema a situacions excepcionals.
- **Precondicions:** suposicions sobre l'estat del sistema abans de la invocació de l'operació.
- **Postcondicions:** canvis d'estat que s'han produït després d'invocar a l'operació.
- **Sortida:** descripció de la sortida que proporciona l'operació.

2.7.1 Interaccions amb model

A continuació especifiquem el contracte de l'operació Moviment dels casos d'ús Avançar resectoscopi, Retrocedir resectoscopi, Girar resectoscopi sentit antihorari, Girar resectoscopi sentit horari, Pivotar resectoscopi amunt, Pivotar resectoscopi avall, Pivotar resectoscopi dreta, Pivotar resectoscopi esquerra i Tallar.

Contracte de l'operació Moviment

Nom: Moviment(xCandidat: Float, yCandidat: Float, zCandidat: Float, angleXCandidat: Float, angleYCandidat: Float, angleHorari: Float): Boolean

Responsabilitats: Permetre a un Cirurgià moure el resectoscopi en qualsevol direcció i tallar en cas que pertoqui.

Excepcions: Si existeix algun paràmetre sense valor o amb valor fora de límit, es desencadena el tractament de l'excepció.

Precondicions: Existeix una instància *r* de resectoscopi, una instància *g* de ganiveta i una instància *i* de Interactuator.

Postcondicions:

Discretitzem el moviment entre la posició candidata i la posició actual de *r* en parts iguals de tamany vòxel i amb les seves components (x, y, z, angleX, angleY, angleHorari) corresponents.

compoActual:= primera component intermitja

Mentre (compoActual != candidata i (la regió a la que correspon compoActual no sigui de tipus pròstata ni tampoc de tipus frontera_pròstata o sigui de tipus frontera_pròstata i g.corrent == 1 i i.força != 0)) **fer**

 Actualitzar els atributs de posicions i angles de *r* amb els valors de compoActual, així com de tots els seus components.

Si g.corrent == 1 **llavors** Actualitzar el vòxel al que correspon compoActual de la pròstata

Fsi

 Avancem compoActual a la següent component intermèdia.

Fmentre

1. **Si** la compoActual no és la component candidata, retorna fals.
2. En cas contrari, retorna cert.

Sortida:

Mostra la nova posició i els angles de *r* a cada pas entremig i el procés de tall en cas que pertoqui.

A continuació especifiquem el contracte de l'operació TraslladarVarettes dels casos d'ús Estendre anella de tall i Retractar anella de tall.

Contracte de l'operació TraslladarVarettes

Nom: TraslladarVarettes(zCandidat: Float): Boolean

Responsabilitats: Permetre a un Cirurgià estendre o retractar l'anella de tall respecte de la cànula del resectoscopi.

Excepcions: Si *zCandidat* no té valor o té valor fora de límit, es desencadena el tractament de l'excepció.

Precondicions: Existeix una instància *r* de Resectoscopi i la seves varetes *v*.

Postcondicions:

Discretitzem el moviment entre la posició candidata i la posició actual de *v* en parts iguals de tamany vòxel.

zActual:= primer valor de *z* entremig

Mentre (la regió a la que pertany el vòxel (*v.posicio.x*, *v.posicio.y*, *zActual*) no sigui de tipus pròstata i *v.posicio.z* != *zCandidat*) **fer**

v.posicio.z := *zActual*.

Avancem a la següent posició intermèdia.

Fmentre

1. **Si** *v.posicio.z* *v* no és *zCandidat*, retorna fals.
2. En cas contrari, retorna cert.

Sortida:

Mostra la nova posició de *v* a cada pas entremig.

A continuació especificuem el contracte de l'operació *AlterarPasCorrent* dels casos d'ús *Activar pas de corrent nivell de tall*, *Activar pas de corrent nivell de cauterització* i *Desactivar pas de corrent*.

Contracte de l'operació *AlterarPasDeCorrent*

Nom: *AlterarPasDeCorrent*(corrent: Integer)

Responsabilitats: Permetre a un Cirurgià activar el pas de corrent a través de l'anella de tall pels nivells de tall o cauteritzar o per altra banda desactivar-lo.

Excepcions: Si *corrent* no té valor o té valor fora de límit, es desencadena el tractament de l'excepció.

Precondicions: Existeix una instància *g* de Ganiveta.

Postcondicions:

1. $g.corrent := corrent$.

A continuació especifiquem el contracte de l'operació Cauteritzar del cas d'ús Cauteritzar.

Contracte de l'operació Cauteritzar

Nom: Cauteritzar(posició: Punt3D): Boolean

Responsabilitats: Permetre a un Cirurgià cauteritzar una ferida situada en un punt determinat.

Excepcions: Si *posició* no té valor o té valor fora de límit, es desencadena el tractament de l'excepció.

Precondicions: Existeix una instància *i* de Interactuador tal que el seu atribut força és igual a 0 i una instància *g* de Ganiveta.

Postcondicions:

1. Si el vòxel al qual pertany la posició *posició* no pertany a la regió frontera_pròstata **i/o** no té l'atribut sang a cert **i/o** $g.corrent == 0$ **i/o** $g.corrent == 1$, retorna fals.
2. En cas contrari, retorna cert i cauteritza el vòxel corresponent actualitzant l'atribut sang a fals.

Sortida:

Mostra el nou estat del vòxel afectat.

A continuació especifiquem el contracte de l'operació InjectarAigua del cas d'ús Injectar Aigua.

Contracte de l'operació Injectar aigua

Nom: InjectarAigua(posició: Punt3D)

Responsabilitats: Permetre a un Cirurgià injectar aigua en una zona determinada per a netejar-la.

Excepcions: Si *posició* no té valor o té valor fora de límit, es desencadena el tractament de l'excepció.

Precondicions: -

Postcondicions:

1. **Si** els vòxels que pertanyen a la zona d'injecció d'aigua expandida des de *posició* pertanyen a la regió grumoll, actualitza els vòxels a les regions corresponents per localització.

Sortida:

Mostra l'eliminació de grumolls de la pròstata en les zones afectades per l'injecció d'aigua.

2.7.2 Gestió d'usuaris i fitxers

A continuació especifico el contracte de l'operació Autenticar del cas d'ús Autenticació.

Contracte de l'operació Autenticar

Nom: Autenticar(nomUsuari: String, password: String): Boolean

Responsabilitats: Permetre a unUsuari carregar la seva pròpia sessió.

Excepcions: Si existeix algun paràmetre sense valor, es desencadena el tractament de l'excepció.

Precondicions: -

Postcondicions:

1. **Si** no existeix cap usuari amb nomUsuari *nomUsuari* i password *password*, retorna fals.
2. En cas contrari, retorna cert i es carrega la sessió de l'usuari amb nomUsuari *nomUsuari*.

A continuació especifico el contracte de l'operació GuardarModel del casos d'ús Guardar Model i Sortir Aplicació.

Contracte de l'operació GuardarModel

Nom: GuardarModel (nom: String, path:String, guardar: Boolean): Boolean

Responsabilitats: Permetre a unUsuari guardar un model en un fitxer.

Excepcions: Si existeix algun paràmetre sense valor, es desencadena el tractament de l'excepció.

Precondicions: La sessió conté una instància *us* d'Usuari.

Postcondicions:

1. **Si** guardar == false, retorna fals.
2. En cas contrari retorna cert i:
 - 2.1 **Si** no existeix una instància *m* de model amb el nom *nom* i path *path*, la crea.
 - 2.2 Associa la instància *us* d'Usuari a *m*.

A continuació especifiquem el contracte de l'operació SortirAplicació del cas d'ús Sortir Aplicació.

Contracte de l'operació SortirAplicació

Nom: SortirAplicació():

Responsabilitats: Permetre a un Cirurgià sortir de l'aplicació.

Excepcions: -

Precondicions: -

Postcondicions:

1. Tanca la sessió actual i surt de l'aplicació.

A continuació especifiquem el contracte de l'operació CanviPassword del cas d'ús Canvi password Cirurgià i Canvi password Administrador en el cas en el que es vulgui canviar la contrasenya pròpia de l'usuari autènticat.

Contracte de l'operació CanviPassword

Nom: CanviPassword (nomUsuari: String, passActual: String, passNova: String): Boolean

Responsabilitats: Permetre a un Usuari canviar la seva pròpia contrasenya.

Excepcions: Si existeix algun paràmetre sense valor, es desencadena el tractament de l'excepció.

Precondicions: -

Postcondicions:

1. **Si** no existeix un Usuari amb el nomUsuari *nomUsuari* i amb password *passActual*, retorna fals.

2. En cas contrari, retorna cert i canvia el password de l'Usuari amb nomUsuari *nomUsuari* per *passNova*.

A continuació especifiquem el contracte de l'operació CanviPassCirViaAdmin del cas d'ús Canvi password Cirurgia en el cas en el que l'administrador realitzi el canvi de la contrasenya d'un cirurgià.

Contracte de l'operació CanviPassCirViaAdmin

Nom: CanviPassCirViaAdmin (nomUsuari: String, passNova: String)

Responsabilitats: Permetre a l'Administrador canviar la contrasenya d'un Cirurgià.

Excepcions: Si existeix algun paràmetre sense valor, es desencadena el tractament de l'excepció.

Precondicions: -

Postcondicions:

1. Canvia el password de l'Usuari amb nomUsuari *nomUsuari* per *passNova*.

A continuació especifiquem el contracte de l'operació CarregarModel del cas d'ús Carregar Model.

Contracte de l'operació CarregarModel

Nom: CarregarModel(model: String, directoriOrigen: String): Boolean

Responsabilitats: Permetre a un Cirurgià carregar un model des d'un directori origen.

Excepcions: Si existeix algun paràmetre sense valor, es desencadena el tractament de l'excepció.

Precondicions: Existeix una instància *c* de Cirurgià.

Postcondicions:

1. Si no existeix un Model amb nom *model* i amb directori *directoriOrigen*, retorna fals.

2. En cas contrari, retorna cert i *c.modelActual* passa a ser la instància de Model amb nom *model* i amb directori *directoriOrigen*.

Sortida:

Mostra c.modelActual

A continuació especifiquem el contracte de l'operació ConsultarAjuda del cas d'ús Consultar Ajuda.

Contracte de l'operació ConsultarAjuda

Nom: ConsultarAjuda()

Responsabilitats: Permetre a un Cirurgià consultar l'ajuda de l'aplicació.

Excepcions: -

Precondicions: -

Postcondicions: -

Sortida:

Mostra l'ajuda de l'aplicació.

A continuació especifiquem el contracte de l'operació AltaCirurgià del cas d'ús Alta Cirurgià.

Contracte de l'operació AltaCirurgià

Nom: AltaCirurgià(nomUsuari: String, password: String): Boolean

Responsabilitats: Permetre a l'Administrador donar d'alta a nou Cirurgià.

Excepcions: Si existeix algun paràmetre sense valor, es desencadena el tractament de l'excepció.

Precondicions: -

Postcondicions:

1. **Si** ja existeix un usuari Cirurgià amb nomUsuari *nomUsuari*, retorna fals.
2. En cas contrari, retorna cert i crea una instància de Cirurgià amb nomUsuari *nomUsuari* i amb password *password*.

A continuació especifiquem el contracte de l'operació BaixaCirurgià del cas d'ús Baixa Cirurgià.

Contracte de l'operació BaixaCirurgia

Nom: BaixaCirurgia(nomUsuari: String): Boolean

Responsabilitats: Permetre a l'Administrador donar de baixa a un Cirurgia.

Excepcions: Si *nomUsuari* no té valor, es desencadena el tractament de l'excepció.

Precondicions: -

Postcondicions:

1. **Si** no existeix un usuari Cirurgia amb nomUsuari *nomUsuari*, retorna fals.
2. En cas contrari, retorna cert i elimina la instància amb nomUsuari *nomUsuari*.

3 Anàlisi: Disseny

3.1 Arquitectura física del sistema

El sistema s'instal·larà en un PC amb sistema operatiu Linux, el qual ha de contenir la llibreria gràfica OpenGL.

L'esquema bàsic d'un sistema de simulació quirúrgica com el que tractem en aquest projecte ve donat per un subsistema de sensors en el qual el cirurgià interactua mitjançant un *haptic* i un altre subsistema de visualització en el qual es mostra per pantalla el procés de simulació efectuat en el subsistema de sensors. La Figura 3.1 il·lustra l'esquema bàsic de les simulacions quirúrgiques. La secció *a* correspon al subsistema de sensors i la secció *b* al subsistema de visualització.

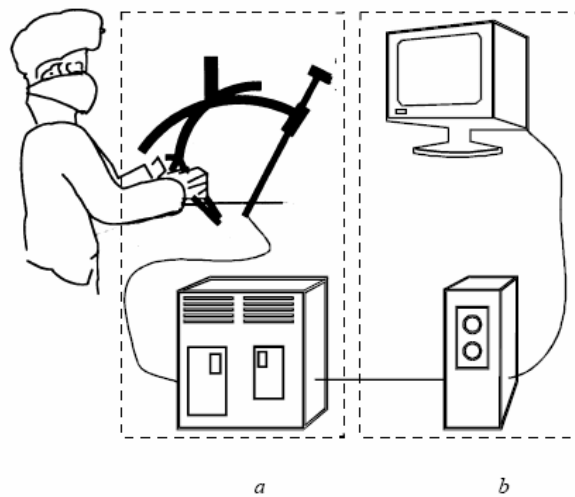


Figura 3.1: Esquema de simulació quirúrgica

El hardware que proporcionarà l'entrada de dades cap a l'aplicació serà un *haptic* amb sis graus de llibertat i amb capacitat per reconèixer la força del moviment. En concret, ens hem decantat pel **Phantom Desktop** de *Sensable Technologies*. Les avantatges d'aquest *haptic* són que és fàcil d'utilitzar i la seva instal·lació no comporta problemes ja que es connecta a través d'un port paral·lel simple. És un hardware que incorpora un disseny totalment portable. A

més, presenta nombrosos indicadors lluminosos per facilitar la seva usabilitat. A continuació detallem algunes especificacions tècniques del Phantom Desktop:

Capacitat de moviment	> 160 W x 120 H x 120 D mm
Mesures	143 W x 184 D mm
Pes	2.86 kg
Rigidesa	Eix X: 1.86 N / mm Eix Y: 2.35 N / mm Eix Z: 1.48 N / mm
Graus de llibertat	Eix x, eix y, eix z, angle amb recolzament x, angle amb recolzament y i angle rotatori z.
Interfície	Port paral·lel
Plataformes compatibles	PC's de tipus Intel
Compatible amb <i>OpenHaptics™ toolkit</i>	Sí

W = Amplada, H = Altura, D = Profunditat, mm= Mil·límetres, kg= Kilograms, N=Newton

La Figura 3.2 correspon a una fotografia del *haptic Phantom Desktop*.



Figura 3.2: Fotografia del *haptic Phantom Desktop*

La selecció d'aquest haptic no ha estat senzilla ja que el mercat actual incorpora d'altres *haptics* de gran qualitat. Primerament ens hem decantat pels models Phantom perquè són els més adequats en quant a tamany i prestacions, a més són uns models punters en el mercat. En concret de models de Phantom hi ha 4 de bastant importants: Phantom OMNi, Phantom Desktop, Phantom Premium 1.5 i Phantom 6DOF. El Phantom OMNi es va descartar ràpidament ja que la seva precisió no és suficient com per realitzar simulacions rigoroses. Per altra banda, Phantom Premium 1.5 i Phantom 6DOF són models molt cars, ocupen molt d'espai i simulen una part del braç totalment innecessària pel nostre projecte. Per contra el Phantom Desktop té una qualitat bastant elevada, és bastant econòmic i portable. Totes aquestes raons han estat suficients com per decidir que el **Phantom Desktop** és el haptic més indicat per realitzar simulacions de cirurgia endoscòpica prostàtica.

3.2 Arquitectura lògica del sistema

Tenint en compte les propietats que afecten a la decisió de l'arquitectura, s'escullen d'una llista de patrons arquitectònics, aquells que millor s'adapten a les necessitats del sistema. Un patró arquitectònic expressa un esquema d'organització estructural fonamental pels sistemes de software. Proporciona un conjunt de subsistemes, especifica les seves responsabilitats i inclou regles i guies per organitzar les relacions entre ells. Els patrons arquitectònics aporten l'esquema general del sistema. Després és necessari adaptar el patró arquitectònic escollit, al sistema concret que s'està desenvolupant.

El patró arquitectònic que s'utilitza en aquest projecte és l'arquitectura en tres capes. Aquestes capes són les següents:

1. Capa de presentació: Serà la que implementarà la interfície amb l'usuari.
2. Capa de domini: Serà la que contindrà la lògica de l'aplicació, estructurant tota la informació continguda en memòria i servint de passarel·la entre les altres dues capes.
3. Capa de gestió de dades: Aquesta capa s'encarregarà d'implementar una interfície entre la capa de domini i les dades emmagatzemades en l'espai personal de disc dur dels usuaris corresponents, de forma que els canvis a les dades no afectin a la capa de domini i viceversa.

El sistema s'implementarà completament en **C++** i **OpenGL**. Aquesta última serà la tecnologia encarregada de la implementació del model prostàtic. Creiem que són les tecnologies que millor s'adapten al tipus d'aplicació que ens ocupa. Per altra banda, per a la implementació de les finestres de la capa de presentació s'ha fet ús de **Qt**, que és una tecnologia especialitzada en implementació d'interfícies per a aplicacions programades en C++.

3.2.1 Disseny de la capa de presentació

En quant al disseny de la capa de presentació, el subdividirem en dues tasques clarament diferenciades: disseny extern i disseny intern. En la part del disseny extern farem la definició de la interacció de l'usuari amb el sistema software. Per tant, la finalitat d'aquesta tasca és dissenyar els elements que l'usuari veurà en l'aplicació. Per altra banda, el disseny intern serà l'encarregat de definir la interacció entre la interfície d'usuari i la capa de domini.

El disseny extern de la capa de presentació està format per una descripció de les finestres que contindrà l'aplicació. Tal i com s'ha explicat anteriorment, la implementació d'aquestes finestres ha estat realitzada amb la tecnologia Qt i estan implementades de manera molt intuïtiva per a l'usuari.

A continuació descriurem les finestres de l'aplicació acompanyant aquesta descripció amb captures de pantalla de les finestres implementades amb Qt.

La Figura 3.3 mostra l'aspecte de la finestra inicial de l'aplicació Simpros en la que l'usuari ha d'autenticar-se per tal de poder accedir a la resta de funcionalitats. La finestra conté el logo descriptiu de l'aplicació. Els únics camps necessaris per a la realització d'aquesta autenticació són un nom d'usuari i una contrasenya que siguin vàlids.

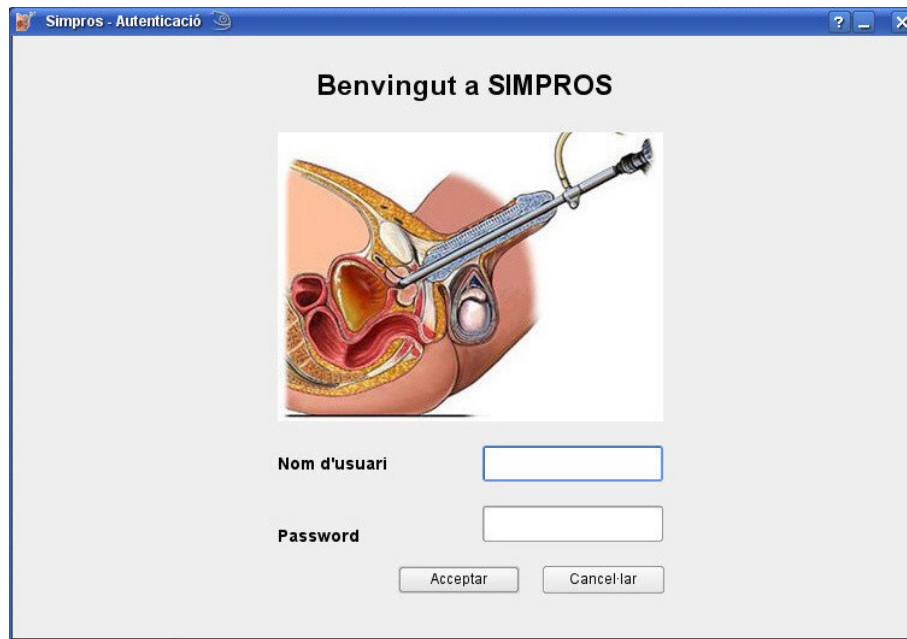


Figura 3.3: Finestra d'autenticació

L'aplicació conté una finestra per mostrar els missatges d'error que poden aparèixer, així com els possibles avisos informatius. La finestra és comuna per a tots els casos d'errors o avisos i el que varia és el text que conté el *label* que és específic per a cada cas i el nom de la finestra que pot ser error o avís. La Figura 3.4 il·lustra aquesta finestra per llançar errors i la Figura 3.5 il·lustra la finestra per llançar avisos informatius.

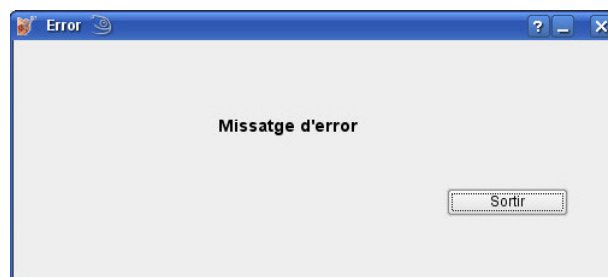


Figura 3.4: Finestra per a missatges d'error

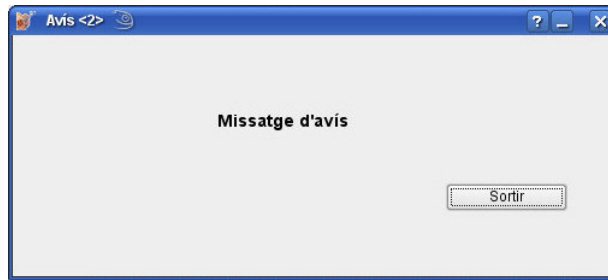


Figura 3.5: Finestra per a avisos informatius

La finestra principal de l'aplicació permet accedir a les diferents funcionalitats de l'aplicació i a més, mostrarà el model prostàtic i permetrà la interacció entre l'usuari i el simulador. La finestra es compon d'una barra de menú amb tres menús: Arxiu, Gestió Usuaris i Ajuda. El menú Arxiu conté les funcionalitats de carregar un model, guardar un model, guardar un model amb un nom determinat i sortir de l'aplicació. En quant al menú Gestió Usuaris hi ha dos opcions depenent de si l'usuari que ha fet login és administrador o cirurgià. En el cas dels cirurgians només té la possibilitat de canviar la seva pròpia contrasenya. Per contra l'administrador pot canviar la seva contrasenya i la dels cirurgians i pot donar d'alta o de baixa cirurgians. Per últim, el menú Ajuda conté una ajuda del funcionament de l'aplicació i una secció pels crèdits de Simpros. La Figura 3.6 il·lustra l'aspecte de la finestra principal de l'aplicació:

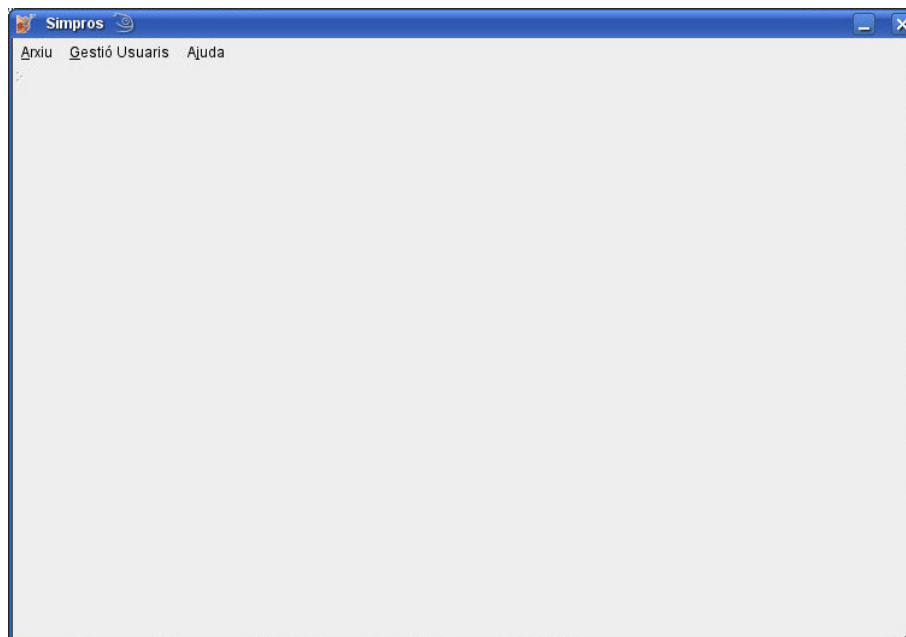


Figura 3.6: Finestra principal

La Finestra per donar d'alta a un nou cirurgià es mostra en la Figura 3.7. Els únics camps necessaris per a la realització d'aquesta funcionalitat són la introducció d'un nou nom d'usuari i d'una contrasenya de manera que l'usuari introduït encara no existeix.

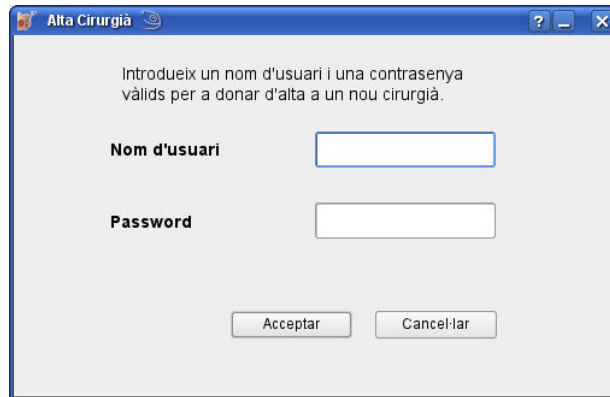
The image shows a Windows-style dialog box titled 'Alta Cirurgià'. It has a blue title bar with a question mark icon, a minimize button, and a close button. The main area is light gray and contains the text: 'Introdueix un nom d'usuari i una contrasenya vàlids per a donar d'alta a un nou cirurgià.' Below this text are two labels: 'Nom d'usuari' and 'Password', each followed by a white text input field. At the bottom of the dialog are two buttons: 'Acceptar' and 'Cancel·lar'.

Figura 3.7: Finestra per a donar d'alta cirurgians

La finestra per donar de baixa a un cirurgià es mostra en la Figura 3.8. L'únic camp necessari per a la realització d'aquesta funcionalitat és la introducció d'un nom d'usuari vàlid.

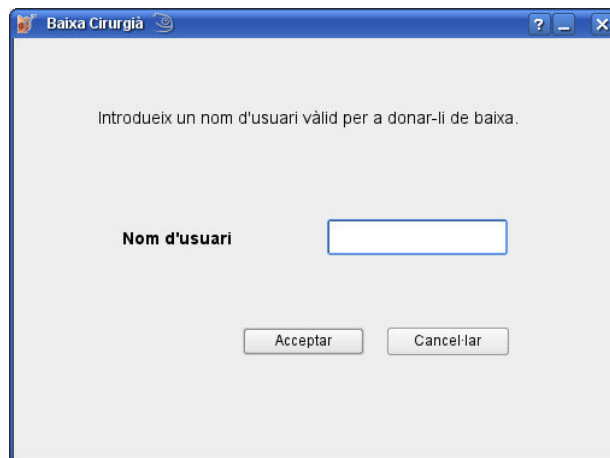
The image shows a Windows-style dialog box titled 'Baixa Cirurgià'. It has a blue title bar with a question mark icon, a minimize button, and a close button. The main area is light gray and contains the text: 'Introdueix un nom d'usuari vàlid per a donar-li de baixa.' Below this text is a single label: 'Nom d'usuari', followed by a white text input field. At the bottom of the dialog are two buttons: 'Acceptar' and 'Cancel·lar'.

Figura 3.8: Finestra per a donar de baixa cirurgians

La finestra per canviar la contrasenya de l'administrador es mostra en la Figura 3.9. L'únic necessari per a la realització d'aquesta funcionalitat és la introducció de la contrasenya actual de l'administrador vàlida i d'una verificació de contrasenyes adequada.

The dialog box is titled "Canviar password Administrador". It contains the instruction: "Introdueix els camps necessaris per a realitzar el canvi de contrasenya d'un usuari administrador." Below this, there are three input fields: "Password actual", "Nova Password", and "Verificació nova Password". At the bottom, there are two buttons: "Acceptar" and "Cancel·lar".

Figura 3.9: Finestra per a canviar la contrasenya de l'administrador

La finestra per canviar la contrasenya personal d'un cirurgià des de la seva pròpia sessió es mostra en la Figura 3.10. L'únic necessari per a la realització d'aquesta funcionalitat és la introducció d'un nom d'usuari i d'una contrasenya actual vàlids i d'una verificació de contrasenyes adequada.

The dialog box is titled "Canviar password Cirurgia". It contains the instruction: "Introdueix els camps necessaris per a realitzar el canvi de contrasenya d'un usuari cirurgià." Below this, there are four input fields: "Nom d'usuari", "Password actual", "Nova Password", and "Verificació nova Password". At the bottom, there are two buttons: "Acceptar" and "Cancel·lar".

Figura 3.10: Finestra per a canviar la contrasenya dels cirurgians

La finestra per mostrar l'ajuda bàsica de l'aplicació en forma de continguts es mostra en la Figura 3.11.

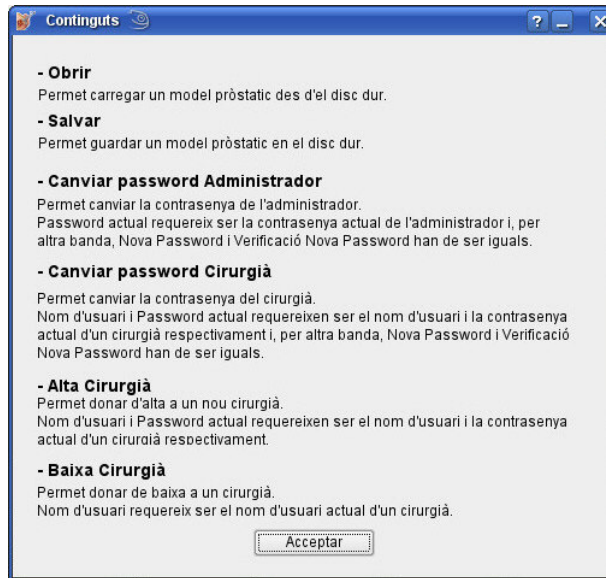


Figura 3.11: Finestra d'ajuda de l'aplicació

La finestra per mostrar informació de l'aplicació Simpros tal i com la versió i l'autor es mostra en la Figura 3.12.

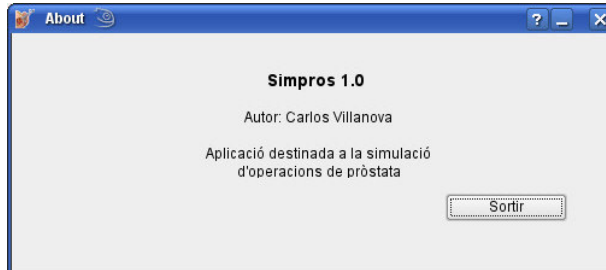


Figura 3.12: Finestra informativa de l'aplicació

El **disseny intern** de la capa de presentació està format per dues parts clarament diferenciades. La primera és el mapa de navegació, que mostra la interacció i navegació entre les diferents finestres de l'aplicació. La segona part és l'encarregada de mostrar la interacció entre les capes de presentació i de domini. Això serà possible amb la realització de diagrames de seqüència totalment il·lustratius.

A continuació mostrem els **mapes de navegació** entre les finestres de l'aplicació pels casos d'ús que comporten més interacció entre finestres:

En la Figura 3.13 il·lustrem el mapa de navegació del cas d'ús autenticar en el que l'usuari accedeix a la finestra principal de l'aplicació si introdueix les dades d'autenticació correctament o es tracta l'error en cas contrari.

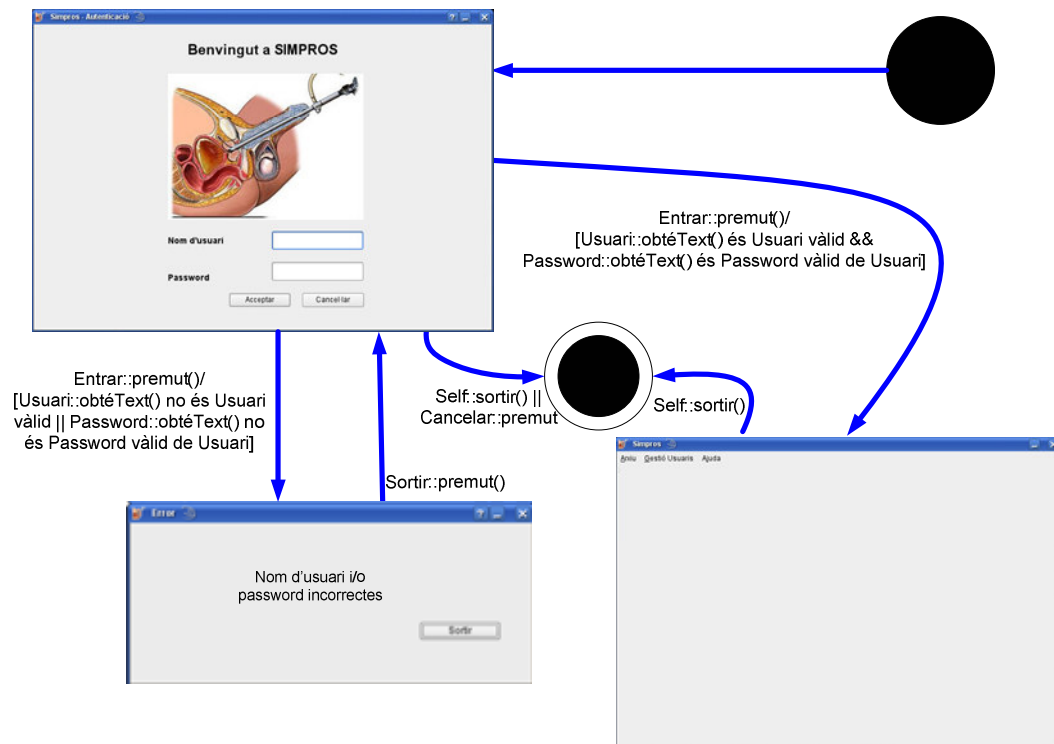


Figura 3.13: Mapa de navegació de l'autenticació

En la Figura 3.14 il·lustrem el mapa de navegació del cas d'ús canvi password administrador en el que l'administrador certifica el canvi de contrasenya si introdueix de manera satisfactòria les dades demanades. Aquestes dades són la contrasenya actual, la nova contrasenya i la verificació d'aquesta nova contrasenya. Si el procés és satisfactori, l'usuari rep un avís d'acceptació en el canvi de password. En cas contrari, es mostra l'error corresponent. És interessant comentar que el cas d'ús canvi password cirurgia comporta un mapa navegacional molt similar a aquest, les diferències són que es requereix incloure el camp nom d'usuari ja que l'administrador pot canviar contrasenyes de diferents cirurgians i en el cas de que el canvi el realitzi l'administrador, no es requereix el camp password actual ja que l'administrador no necessita conèixer la contrasenya actual del cirurgià per tal d'efectuar el canvi.

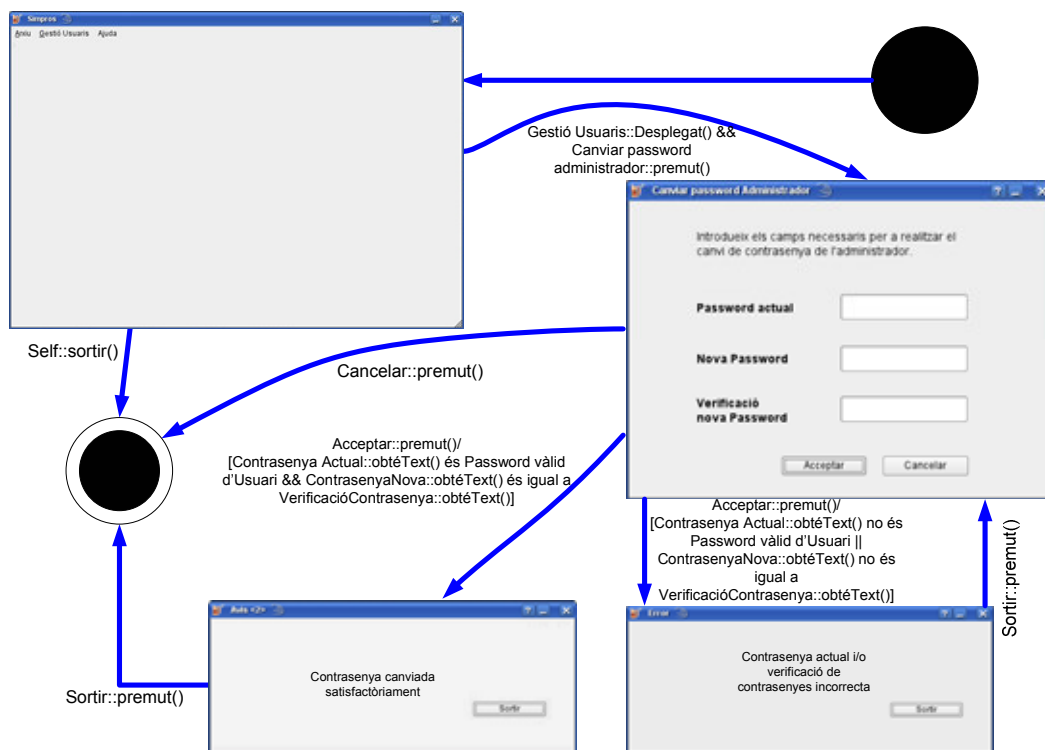


Figura 3.14: Mapa de navegació del canvi de password d'administrador

En la Figura 3.15 il·lustrem el mapa de navegació del cas d'ús Alta cirurgia en el que l'administrador dona d'alta a un cirurgià si el camp nom d'usuari no correspon a cap usuari existent i password està omplert de manera correcte. En el cas de que es doni d'alta a un nou cirurgià, el sistema mostrarà un avís informatiu. En cas contrari, es mostrarà l'error corresponent.

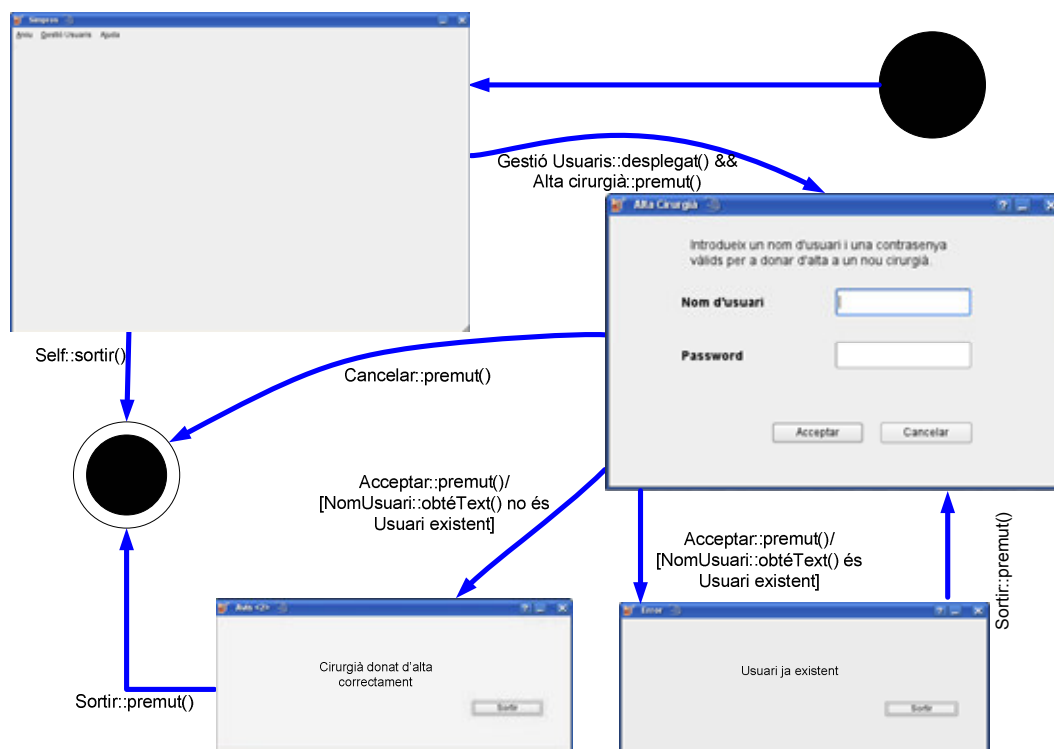


Figura 3.15: Mapa de navegació de l'alta de cirurgians

En la Figura 3.16 il·lustrem el mapa de navegació del cas d'ús Baixa cirurgia en el que l'administrador dona de baixa a un cirurgià si el camp nom d'usuari correspon a un usuari existent. En el cas de que es doni de baixa a un cirurgià, el sistema mostrarà un avís informatiu. En cas contrari, es mostrarà l'error corresponent.

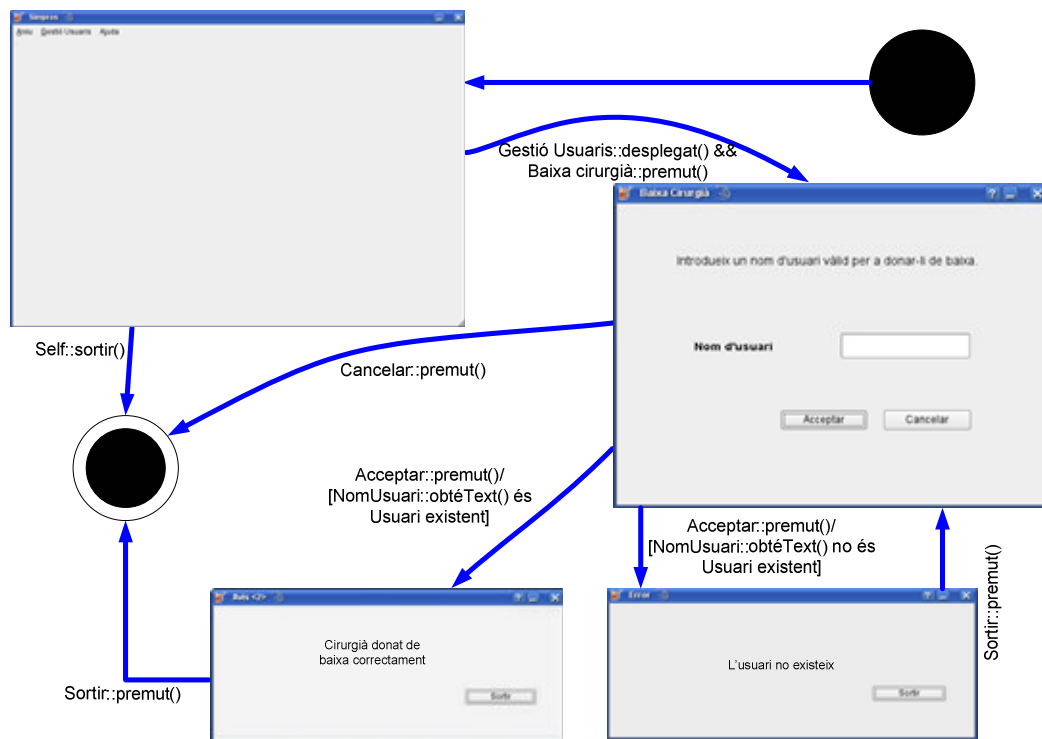


Figura 3.16: Mapa de navegació de la baixa de cirurgians

A continuació mostrem els diagrames de seqüència que il·lustren la interacció entre les dues capes esmentades anteriorment pels casos d'ús més rellevants. El diagrama de seqüència pel cas d'ús Canvi password Administrador és pràcticament igual al de Canvi password Cirurgià. La Figura 3.17, la Figura 3.18 i la Figura 3.19 descriuen aquests diagrames de seqüència pels casos d'ús Autenticació, Canvi password Cirurgià (en el cas en el que el canvi el realitzi el propi cirurgià) i Alta Cirurgià respectivament:

Autenticació:

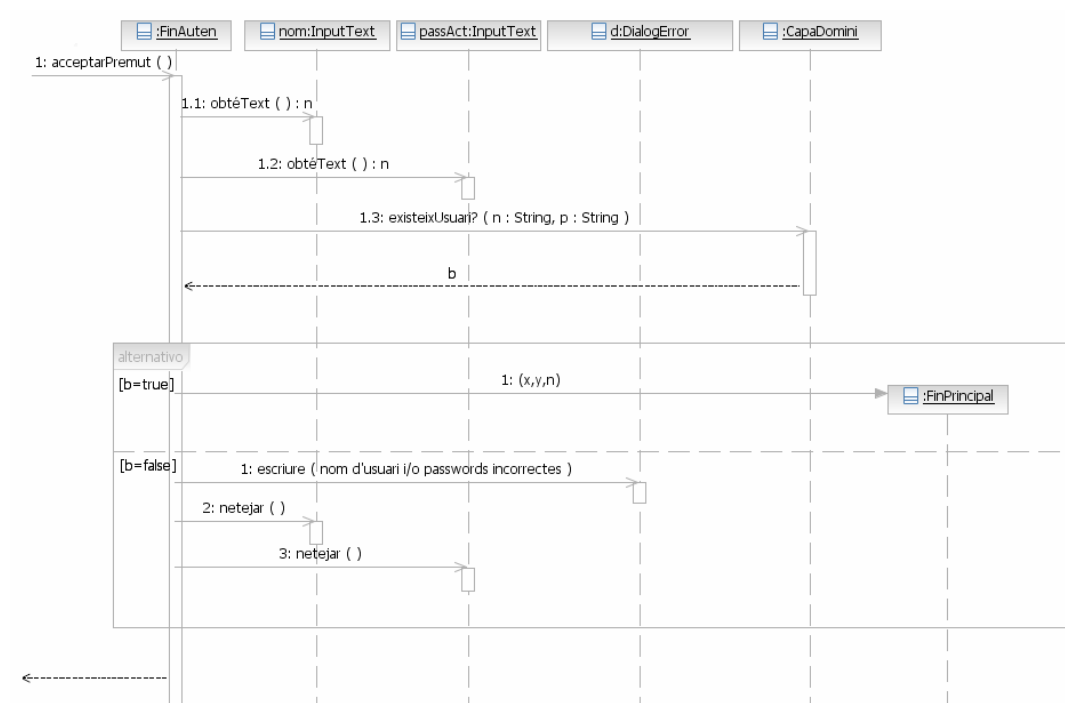


Figura 3.17: Diagrama de seqüència de l'acció d'autenticar

[illegible]

69

Alta Cirurgia:

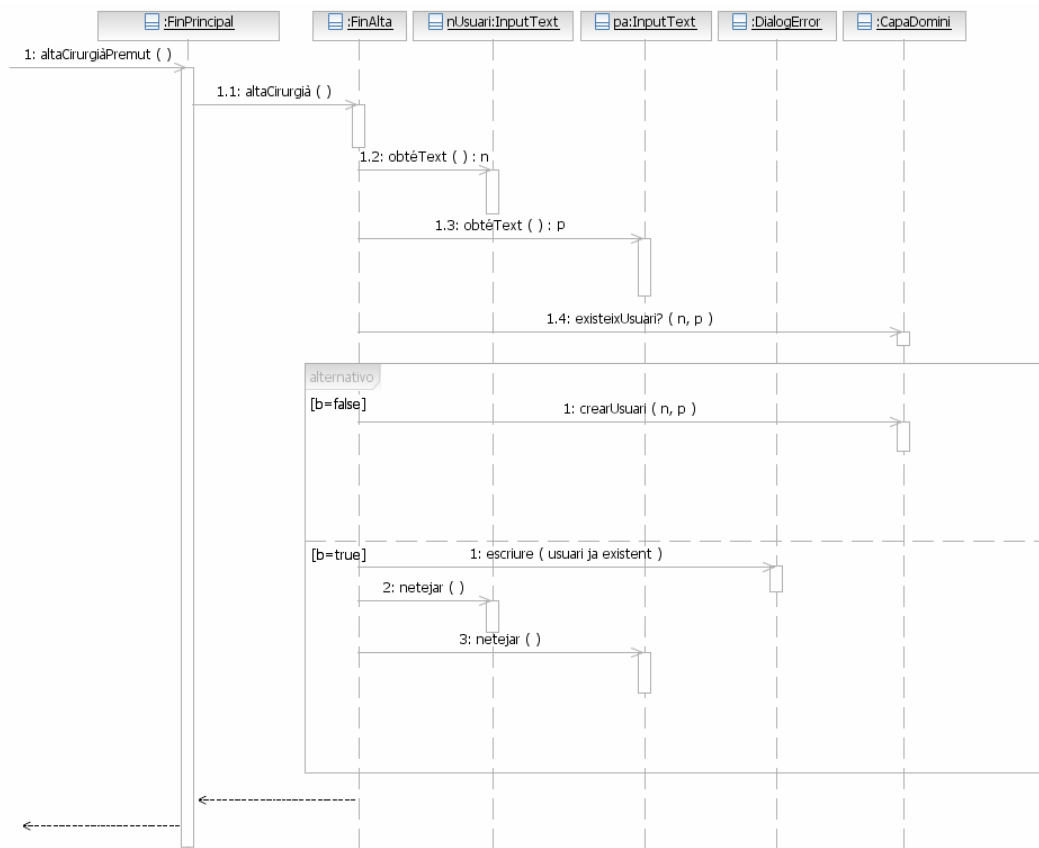


Figura 3.19: Diagrama de seqüència de l'acció de donar d'alta un cirurgià

Existeixen alguns diagrames de seqüència que defineixen operacions utilitzades en els diagrames anteriors. Les figures 3.20, 3.21, i 3.22 il·lustren aquests diagrames de seqüència:

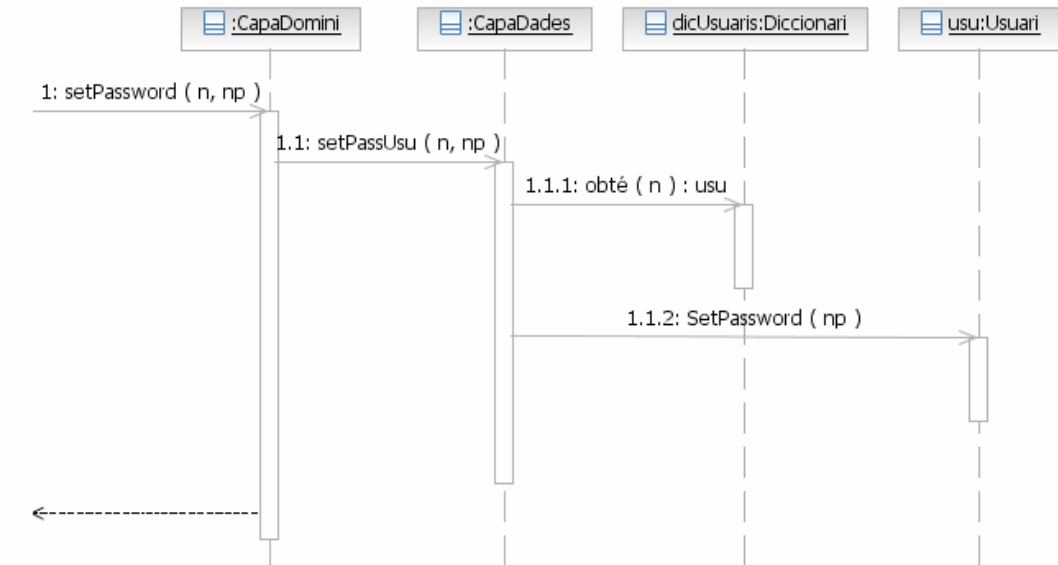


Figura 3.20: Diagrama de seqüència de l'operació `setPassword` de la capa de domini

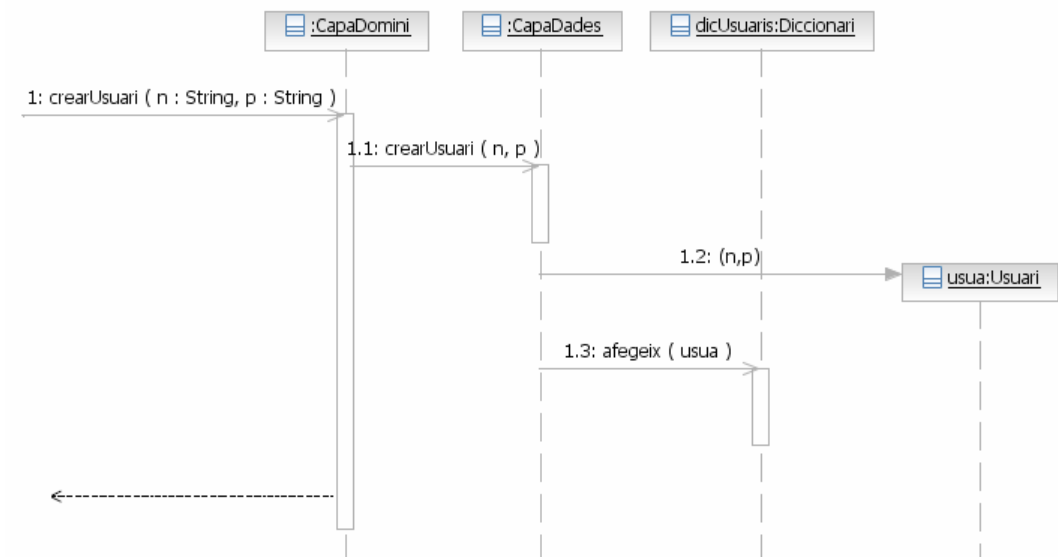


Figura 3.21: Diagrama de seqüència de l'operació `crearUsuari` de la capa de domini

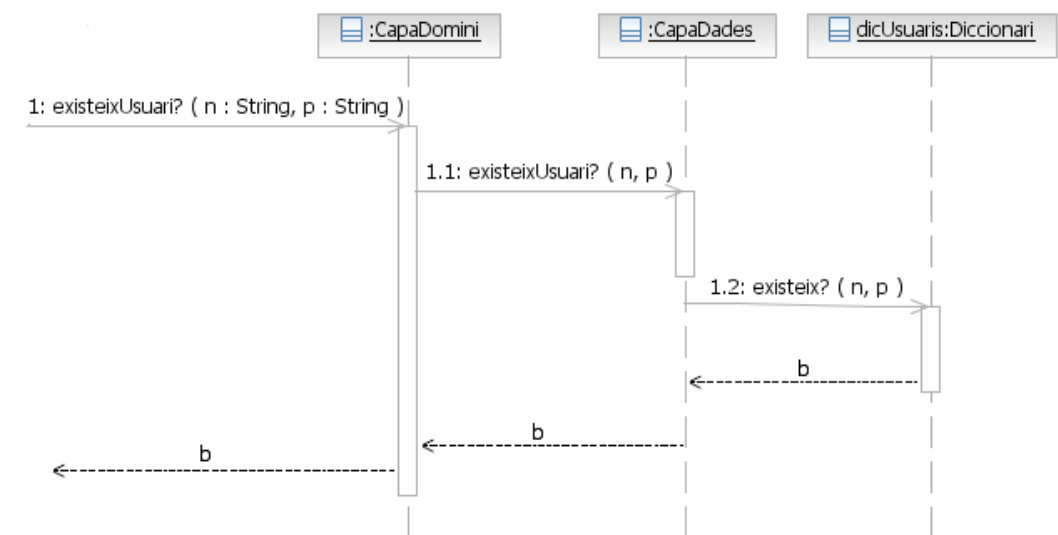


Figura 3.22: Diagrama de seqüència de l'operació existeixUsuari? de la capa de domini

Per tal de dissenyar el **Controlador de la Capa de Presentació**, hem decidit no utilitzar cap dels tipus bàsics de controladors (façana, cas d'ús i transacció) ja que no s'adapten de manera correcta a les nostres necessitats. La idea és separar els controladors per dos subsistemes que seran les dues pantalles on poden iniciar-se requisits funcionals. Aquestes dues pantalles són la pantalla d'autenticació del sistema i la pantalla principal de l'aplicació. La Figura 3.23 il·lustra el disseny d'aquest controlador definit per subsistemes.



Figura 3.23: Disseny del Controlador de la Capa de Presentació

3.2.2 Disseny de la capa de domini

El disseny de la capa de domini depèn de l'aplicació de patrons de disseny que faciliten la posterior implementació de l'aplicació. Aquests patrons de disseny s'utilitzen de manera particular per a cada situació concreta. En aquest projecte s'ha utilitzat el següent patró de disseny per a la capa de domini:

- patró controlador

Aquest disseny també està subdividit en dues tasques. La primera consisteix en la realització del model de disseny del diagrama de classes. Aquest model de disseny conté les classes de la capa de domini, la relació entre elles, els seus atributs i les funcions de cadascuna d'elles. Per altra banda, l'altra tasca del disseny de la capa de domini consisteix en la resolució de les funcionalitats de la capa de domini dins de l'aplicació mitjançant la realització de diagrames de seqüència.

Abans de començar a dissenyar els diagrames de classes del domini, dissenyarem les **regions** possibles a les quals pertanyeran els vòxels del model. Tal i com mostrarem més endavant, les regions seran simplement números enters que representaran zones determinades. La següent taula descriu les regions del model:

Identificador	Nom de la regió	Descripció
1	buida	Regió exterior del model
2	frontera_pròstata	Frontera de la pròstata
3	pròstata	Massa prostàtica interior
4	canal	Canal de la uretra
5	anella	Anelles que separen el canal de la uretra de l'escorça
6	grumoll	Grumoll tallat
7	vareta	Vareta del resectoscopi
8	ganiveta	Ganiveta del resectoscopi
9	extrem_ganiveta	Superfície de tall
10	cànula	Cànula del resectoscopi
11	tallada	Zona tallada, instant abans de l'extracció
12	escorça	Escorça de la pròstata

El diagrama de classes del model de disseny es mostra en parts clarament diferenciades. A continuació en la Figura 3.24 observem la relació entre la classe del **model prostàtic** i la classe que representa el **interactuator** que s'encarrega de rebre els events proporcionats pel cirurgià des del *haptic*. Per tant, emmagatzema atributs tals com la força exercida sobre el

haptic, la posició candidata a ser traslladat el resectoscopi i la component de l'eix z candidata a traslladar les varetes. Per altra banda, la classe **visualitzador** és l'encarregada de mostrar gràficament la situació dels components del model prostàtic.

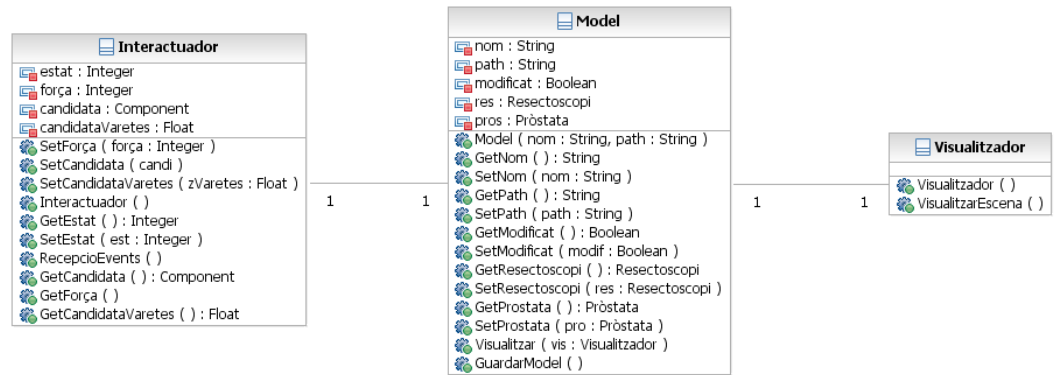


Figura 3.24: Diagrama de classes domini1

El següent subdiagrama il·lustrat a la Figura 3.25 mostra en alt nivell els components del model prostàtic, que són les classes que representen el **resectoscopi** i la **pròstata**. A més, observem que la classe *Impacte*, que era associativa en el model conceptual de l'anàlisi, ha deixat de ser-ho al aplicar-hi disseny.



Figura 3.25: Diagrama de classes domini2

La Figura 3.26 il·lustra com la classe Resectoscopi es compon de varis components necessaris per realitzar el procés de tall de manera correcta. Aquestes parts són la **il·luminació**, les dues **varetes** que sostenen la ganiveta, la pròpia **ganiveta** i la **cànula** del resectoscopi.

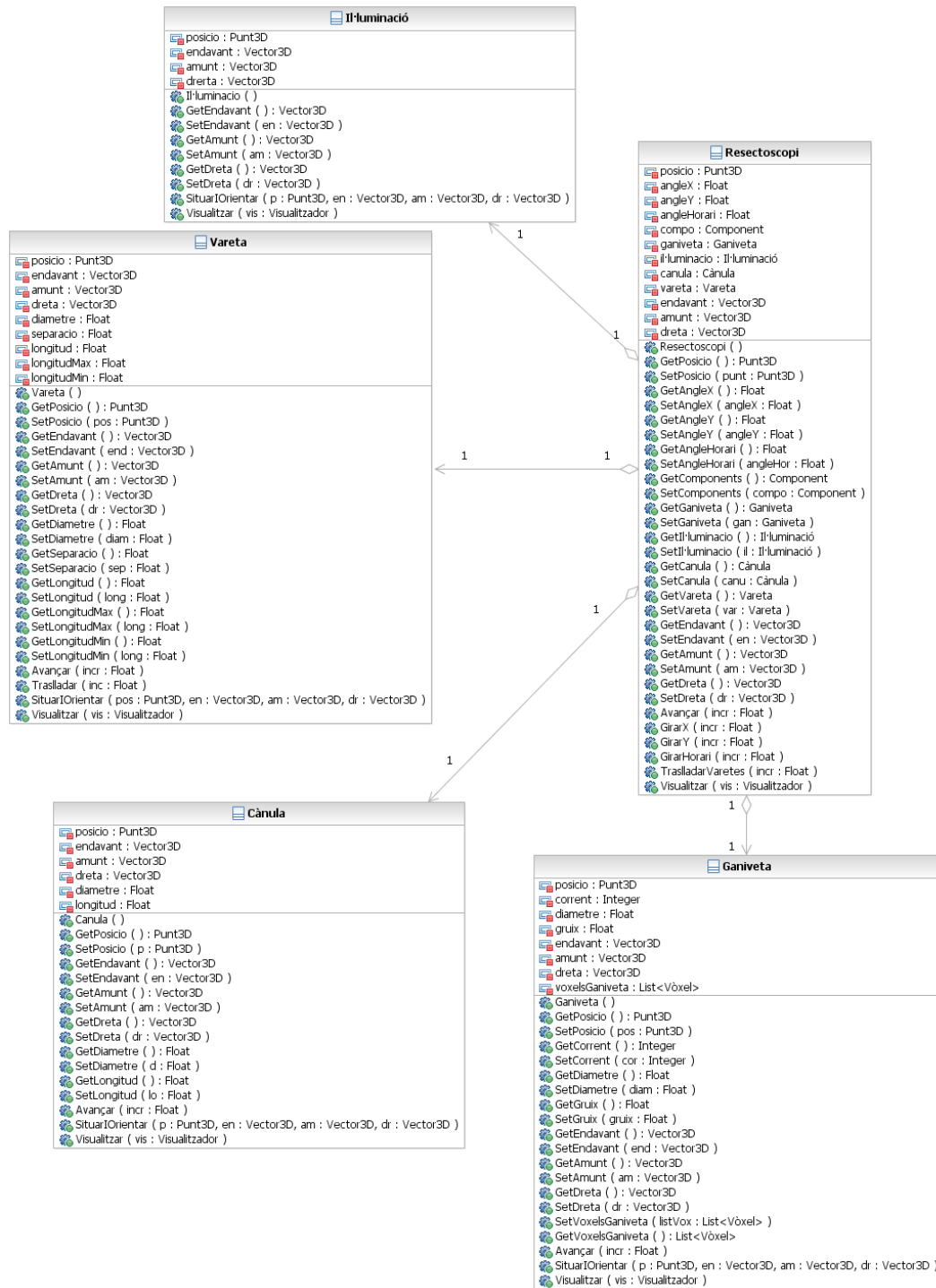


Figura 3.26: Diagrama de classes domini3

La Figura 3.27 il·lustra com la classe que representa la **massa prostàtica** està composta de molts **vòxels**. Cadascun d'aquests vòxels es situa en una posició tridimensional determinada i pertany a una regió concreta.

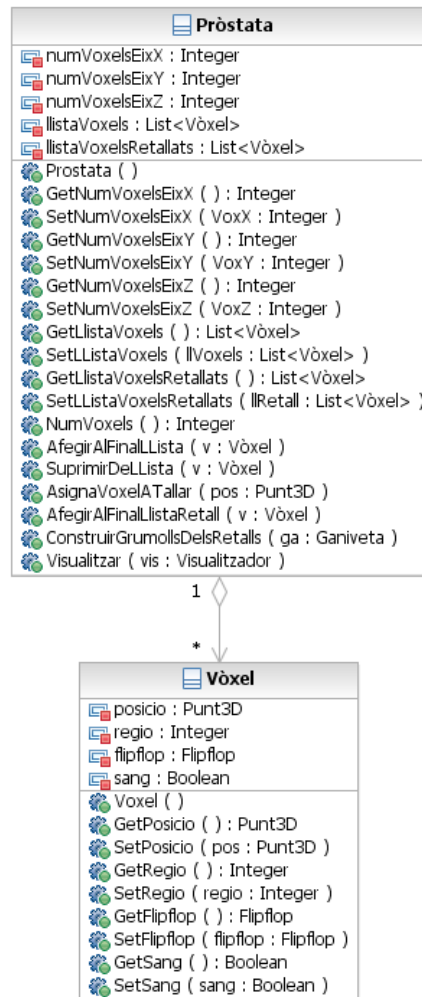


Figura 3.27: Diagrama de classes domini4

La capa de domini es basa principalment en realitzar de manera correcta els moviments del resectoscopi que són generats pel cirurgià. És per aquesta raó que amb un diagrama de seqüència en alt nivell mostrarem l'esquema a seguir per part d'aquesta capa. Com ja hem explicat anteriorment, la classe Interactuador s'encarrega de rebre i processar els events que entren en el sistema des de el *haptic*. Per tant, la seva feina és entendre el tipus de **moviment realitzat** i realitzar-lo progressivament mentre sigui possible. Durant els passos intermitjos d'aquest moviment, s'ha d'actualitzar el resectoscopi i els seus subcomponents i tots els vòxels de la massa prostàtica que s'hagin vist afectats pel moviment o tall. Com que això és un diagrama de seqüència en alt nivell hi ha detalls interns que no apareixen, com per exemple que les actualitzacions de posicions i angles dels components que pertorquin es realitzaran si i

només si estan dins dels límits físics. La Figura 3.28 mostra el diagrama de seqüència en alt nivell d'aquesta operació que controla els moviments realitzats pel cirurgià:

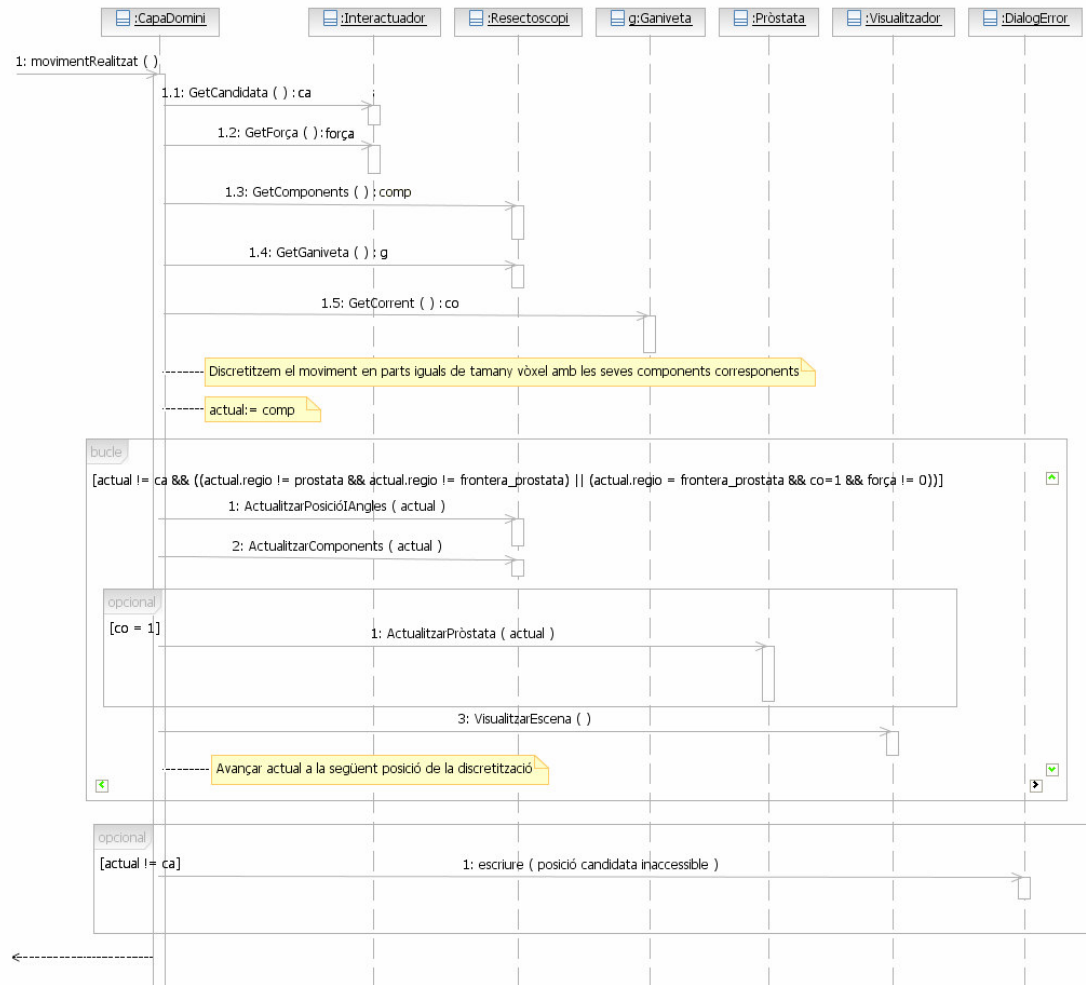


Figura 3.28: Diagrama de seqüència dels moviments en alt nivell

El següent grup de moviments possibles a tractar són els de **trasllats de les varetes**. Hem de recordar que hi ha un límit físic en aquest tipus de moviments. Per altra banda, aquest tipus de moviments també poden combinar-se amb processos de tall de la massa prostàtica. Tal i com hem esmentat abans, hi ha detalls interns que no apareixen en aquest diagrama, com per exemple que les actualitzacions de posicions i angles dels components que pertorquin es realitzaran si i només si estan dins dels límits físics. La figura 3.29 il·lustra aquests moviments possibles que combinen extensions i redreçaments de les varetes amb possibles talls de la massa prostàtica.

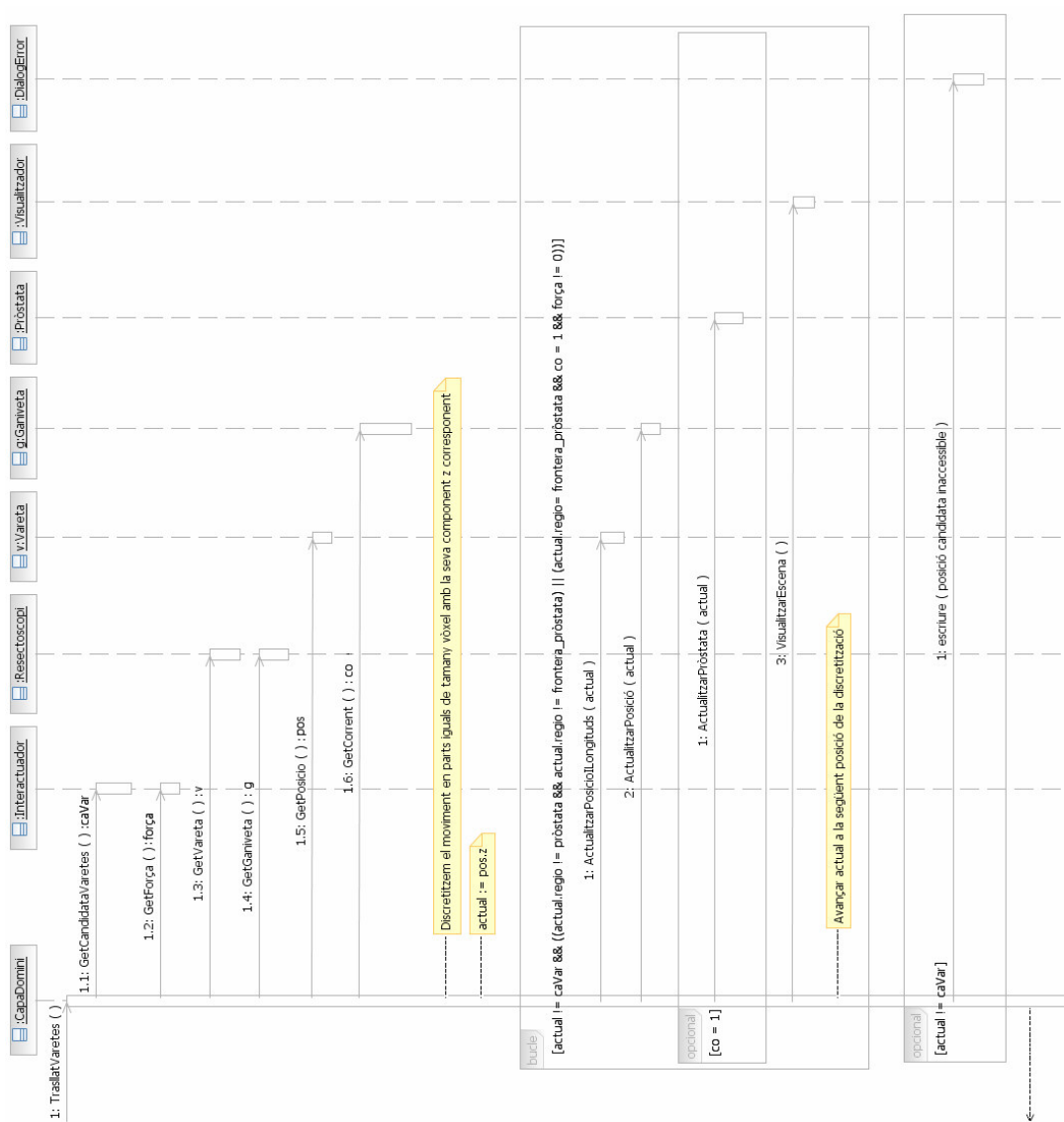


Figura 3.29: Diagrama de seqüència dels trasllats de varetes en alt nivell

Un altre possibilitat és el fet de **cauteritzar** una zona determinada de la pròstata normalment per l'aparició d'una ferida. En aquest cas les premisses són bastant clares, la força exercida ha de ser pràcticament nul·la i superficial, el nivell de corrent ha de ser de tipus de cauterització i per a facilitar la feina s'actualitzarà un atribut que indica si el vòxel està sagnant o no. Per a poder realitzar aquesta acció, prèviament hem de recollir els vòxels que pertanyen a la posició física de la ganiveta. Això es calcula mitjançant la posició, el diàmetres i el gruix de la ganiveta. Per altra banda, hem de assegurar que els vòxels que volem cauteritzar pertanyen a la regió de la frontera de la pròstata. La Figura 3.30 il·lustra el diagrama de seqüència de l'acció de cauteritzar en alt nivell.

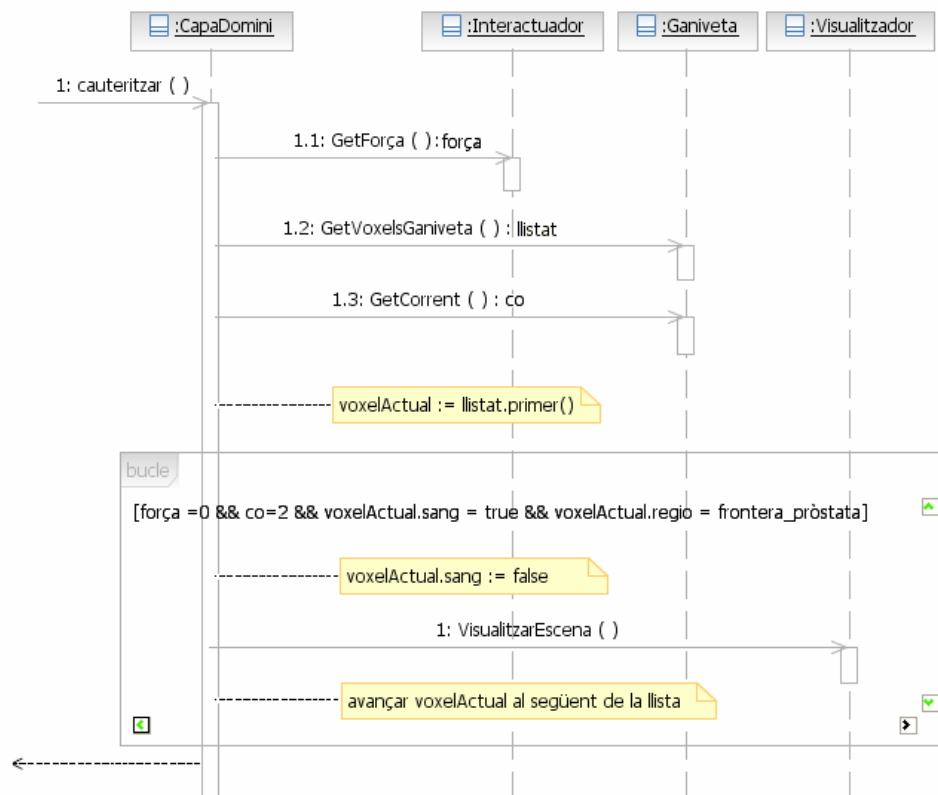


Figura 3.30: Diagrama de seqüència de la cauterització en alt nivell

El **Controlador de la capa de domini** conté les operacions dissenyades en els diagrames de seqüència anteriors a més de l'operació **alterarPasDeCorrent**, que és l'operació encarregada de modificar el tipus de corrent de l'anella de tall o desactivar-lo. Aquesta operació no està dissenyada en els diagrames de seqüència anteriors ja que és totalment trivial. El controlador

de la capa de domini és de tipus façana i, per tant, incorpora tantes operacions com esdeveniments ha de capturar el sistema. La raó per la qual ens hem decantat per aquest tipus de controlador és perquè hi ha un número reduït d'operacions i no tenen cap subdivisió clarament definida. La Figura 3.31 il·lustra el disseny del controlador de la capa de domini.

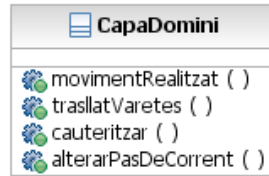


Figura 3.31: Controlador de la capa de domini

3.2.3 Disseny de la capa d'accés a dades

En quant al disseny de la capa de gestió de dades hem decidit prescindir de la utilització d'una base dades ja que pel tipus de informació que volem emmagatzemar no és necessari.

Tal i com hem vist anteriorment, la nostra aplicació requereix d'un administrador que serà la persona indicada de crear les sessions d'usuari i de realitzar les particions personals de disc per a que els usuaris puguin emmagatzemar els seus models prostàtics personals. En un fitxer només editable per l'administrador estarà emmagatzemat de manera segura el llistat de noms d'usuaris i contrasenyes dels cirurgians i el de l'administrador. Aquest fitxer s'anomena *usuaris.txt*. Aquest fitxer és llegit i editat des de l'aplicació. Les contrasenyes estan encriptades de manera que és gairebé impossible conèixer el seu veritable valor. L'algoritme d'encriptació utilitzat ha estat l'obtenció del codi ASCII per a cadascuna de les lletres de les contrasenyes i la suma de la constant 322 en el cas d'un codi parell o la suma de la constant 414 en el cas d'un codi imparell. La codificació no és impossible de descodificar però sí que és cert que és complicadíssim d'esbrinar fàcilment el contingut de les contrasenyes.

En el disseny de la capa d'accés a dades apareixen els següents patrons de disseny:

- patró iterador (cas diccionari)
- patró controlador

El model conceptual d'aquesta capa representa bàsicament els dos possibles usuaris del sistema (Cirurgià i Administrador). A més s'observa que un Cirurgià pot disposar de diversos models prostàtics a la seva sessió d'usuari. La Figura 3.32 il·lustra el diagrama de classes de la capa de dades:

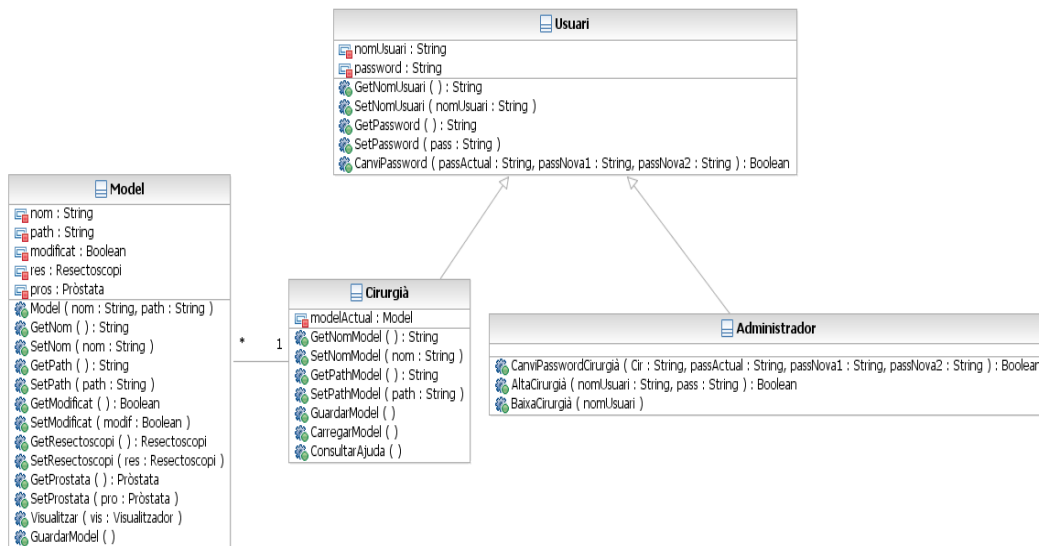


Figura 3.32: Diagrama de classes capa de dades

L'aplicació del patró iterador (cas diccionari) en aquest disseny facilita que es pugui accedir a un agregat de manera senzilla mitjançant una clau. En aquest projecte considerem l'aparició de dos diccionaris, un per a la classe Model i un altre per a les dades dels usuaris. En el primer cas aconseguim que l'accés als models prostàtics dels cirurgians es realitzi de manera eficient i en l'altre cas proposem una manera de recórrer el nostre fitxer de dades d'usuari de manera ràpida i intuïtiva.

A continuació mostrem en la Figura 3.33 una explicació de les funcions que incorpora el iterador de tipus diccionari:

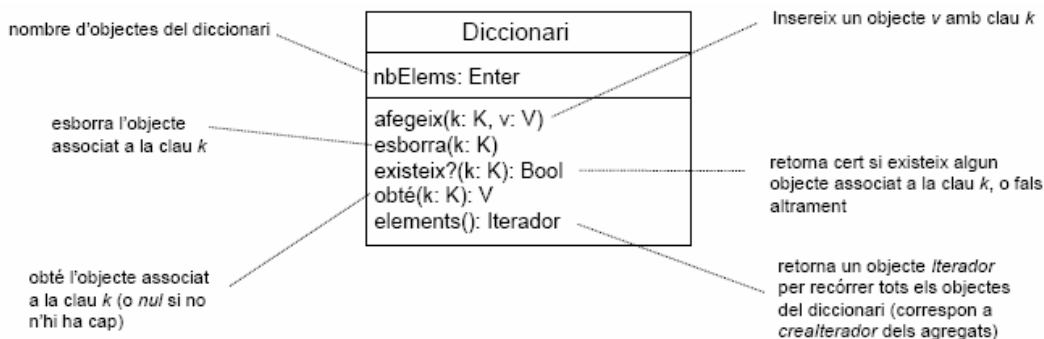


Figura 3.33: Patró iterador (cas Diccionari)

Els diagrames de seqüència aprofiten les avantatges dels diccionaris per a poder obtenir, modificar o esborrar els elements que pertorquin dins de la col·lecció. En concret en la Figura 3.34 veiem com la capa de dades treballa de manera eficient en aquest sentit. En aquest exemple obtenim un model prostàtic concret a partir del seu *path*. Un cop tenim el model ja podem treballar directament sobre ell.

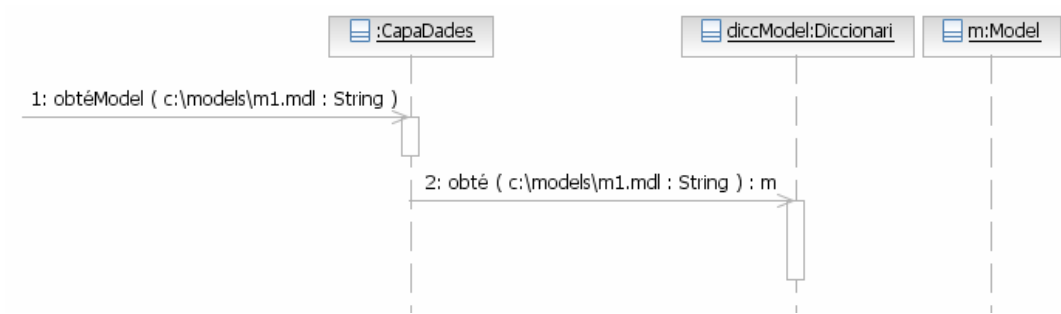


Figura 3.34: Diagrama de seqüència de diccionari de models

4 Implementació

Tal i com s'ha esmentat anteriorment, la única part implementada durant aquest projecte ha estat la interfície gràfica d'usuari de l'aplicació. Per tant, en aquest apartat farem una explicació detallada del procés realitzat en aquesta implementació. A més, farem una descripció de l'entorn de treball i de les eines utilitzades durant el projecte.

4.1 Descripció de l'entorn de treball

El treball realitzat durant aquest projecte ha estat desenvolupat en tres localitzacions diferents. La gran majoria del treball s'ha realitzat a casa meua en la qual treballa amb un Pentium IV a 2,4GHz amb 1 GB de RAM i un sistema operatiu Windows XP Service Pack 2. També vaig estar realitzant diverses sessions de simulació amb la antic simulador prostàtic durant la primera fase del projecte en el campus sud de la UPC. Allà vaig disposar d'un HP XW400 Workstation Intel Core 2 Duo a 2,4GHz x2 amb 2GB RAM, una tarjeta de vídeo NVIDIA Quadro FX 3500 i un sistema operatiu Linux Debian 2.6. Finalment, la implementació de la interfície gràfica la he realitzat en el campus nord de la UPC ja que disposava de tot el necessari per a treballar satisfactòriament amb Qt. En aquest cas disposava de un Intel Pentium IV a 3,2 GHz amb 512MB RAM.

El software utilitzat per a la realització d'aquest projecte ha estat:

- Microsoft Word 2002, el qual he utilitzat per redactar la documentació del projecte.
- Microsoft Project Professional 2003, el qual he utilitzat per realitzar la planificació i seguiment del projecte.
- Microsoft Visio Professional 2003, el qual he utilitzat per realitzar alguns diagrames.
- IBM Rational Software Modeler 7.0, el qual he utilitzat per realitzar la gran majoria de diagrames de l'especificació i del disseny de l'aplicació.
- Simpros, vaig fer diverses sessions de simulació sobre la versió antiga de l'aplicació.
- Qt Designer, el qual he utilitzat per dissenyar les finestres amb la tecnologia Qt.

4.2 Tecnologies utilitzades

En aquest apartat explicarem el significat purament tècnic de la tecnologia Qt, l'ús d'aquesta tecnologia dins de l'aplicació i les avantatges que aporta la seva utilització en el nostre projecte.

4.2.1 Qt

Qt és una llibreria multiplataforma per desenvolupar interfícies gràfiques d'usuari. Va ser creada per la companyia noruega Trolltech. Qt és utilitzada en KDE, un entorn d'escriptori per a sistemes com GNU/Linux o FreeBSD, entre altres. Utilitza el llenguatge de programació C++ de forma nativa i, a més, existeixen bindings per a altres tipus de llenguatges.

Donat que s'ha fet un estudi i s'ha arribat a la conclusió que el més òptim per a que el simulador gràfic treballi en les millors condicions és utilitzar la llibreria gràfica OpenGL, això va implicar que el llenguatge de programació que millor s'adaptava a aquesta llibreria era C++. Qt és totalment compatible amb el llenguatge C++ i amb la llibreria gràfica OpenGL. Per altra banda, cal dir que el dissenyador de finestres de Qt anomenat *Qt Designer*, facilita molt la feina al programador i permet realitzar finestres d'una manera molt intuïtiva. Per totes aquestes raons, la tecnologia que s'ha identificat com a òptima per a la implementació de la capa de presentació ha estat Qt. Hem considerat que era la més convenient per poder deixar un treball fet de cara a la total implementació de l'aplicació en un futur i que no comporti gens de problemes a l'hora treballar a partir del que s'ha fet.

En el cas de la nostra aplicació, tal i com hem descrit en el disseny de la capa de presentació, hem elaborat una sèrie de finestres amb Qt i hem proporcionat lectures i escriptures sobre el fitxer que emmagatzema les dades principals dels usuaris de l'aplicació. El tractament del fitxer descrit en el disseny de la capa de dades és necessari per a la realització d'autenticacions, canvis de contrasenyes o altes/baixes de cirurgians. Per tant, tenint en compte que la capa de domini no està implementada en aquest moment, s'ha realitzat una connexió directa entre les capes de presentació i de dades. Un cop s'hagi implementat tot, aquesta transmissió de dades haurà de passar prèviament pel domini.

A continuació mostrem la secció del codi que s'encarrega de controlar les autenticacions dels usuaris. Aquest codi es pot trobar en el fitxer *autenticacio.ui.h*.

```
#include <qapplication.h>
#include <qaction.h>
#include <qmenubar.h>
#include <qpopupmenu.h>
#include <qtoolbar.h>
#include "principal.h"
#include "errors.h"
#include <iostream>
#include <fstream>
#include <string.h>
#include <sstream>

using namespace std;

/* funció per a autenticar als usuaris, s'executa en el moment en el que els usuaris premen el botó acceptar en la
finestra d'autenticació. */

void Autenticacio::autenticar()
{
    // Emmagatzemem el valor del nom d'usuari introduït per l'usuari
    string lineNom= (this->lineNom->text());
    // Emmagatzemem el valor de la contrasenya introduïda per l'usuari
    string linePass= (this->linePass->text());

    // Obrir el fitxer amb les dades d'usuari per a lectura
    ifstream f("usuaris.txt", ifstream::in);

    string cadena;
    string *noms= new string[50];
    string *pass= new string[50];

    // Realitzem una lectura del fitxer
    f >> cadena;

    while (!f.eof())
    {
        f >> cadena;
    }

    f.close();

    string copia(cadena);
    string copia2(cadena);
    string bona(cadena);
    string resta(cadena);
    string aux;
    int k;
    int i=0;
    int j=0;
    int m=0;

    // mentre no arribem al caràcter finalitzador del fitxer, l'anem llegint
    while(resto.compare("#")!=0){
        /*anem eliminant els separadors d'usuaris del fitxer i posteriorment guardem els noms d'usuaris i
        les contrasenyes codificades per separat*/
        aux=resta;
        k=resta.find("$",0);

        bona = aux.erase(k);
        resto= resta.substr(k+1);

        if(i%2==0){
            noms[j]=bona;
            j++;
        }
        else{
            pass[m]=bona;
```

```

        m++;
    }
    i++;
}

string pa;
string *passw = new string[50];
string q;
bool bol=true;
string s2;
int s3;
string str;
stringstream out;
bool bo=false;
int d=0;

//comprovem que el nom d'usuari introduït per l'usuari coincideixi amb algun dels usuaris del fitxer
while(d<=j && bo==false){
    if(noms[d].compare(lineNom)==0){
        bo=true;
    }
    else d++;
}

if(bo) {
    pa=pass[d];

    int u=0;
    //emmagatzemem les contrasenyes del fitxer eliminant el caràcter separador
    for(int a = 0; a!=pa.length(); ++a){
        q=pa[a];
        if(q.compare(",")!=0){
            passw[u].push_back(pa[a]);
        }
        else u++;
    }

    int z=0;

    if(u==(linePass.length() -1)){
        while(bol==true && z<=u){
            str.clear();
            s3=int(linePass.at(z));

            s2=passw[z];
            out.seekp (ios_base::beg);

            /* Hem de tenir en compte que les contrasenyes estan codificades de manera
            que cadascuna de les lletres estan en codi ascii però, a més, els codis parells
            estan sumats a una constant 322 i els imparells a 414 */
            if(s3%2==0){
                out << s3+322;
            }
            else{
                out << s3+414;
            }

            str = out.str();
            // Comparem la contrasenya introduïda amb la descodificada del fitxer
            if(str.compare(s2)!=0){
                bol=false;
            }
            z++;
        }
    }
    else{
        bol=false;
    }

    if(bol){
        //Aquest és el cas d'usuari administrador identificat correctament
        if(d!=0){
            Simpros *s = new Simpros();

```



```

        s->gesti_usuarisAlta_CirurgiaAction->setVisible(false);
        s->gesti_usuarisCanvi_password_AdministradorAction->setVisible(false);
        s->gesti_usuarisBaixa_CirurgiaAction->setVisible(false);
        s->show();
        this->hide();
    }

    //Aquest és el cas d'usuari cirurgià identificat correctament
    else{
        Simpros *s = new Simpros();
        s->show();
        this->hide();
    }
}

//Aquest és un cas d'autenticació incorrecta
else{
    Errors *e = new Errors();
    e->errorLabel->setText("<font size='\"+1\"><font size='\"+1\">Nom d'usuari i/o contrasenya incorrectes.</font></font>");
    e->show();
}

}

//Aquest és un cas d'autenticació incorrecta
else{
    Errors *e = new Errors();
    e->errorLabel->setText("<font size='\"+1\"><font size='\"+1\">Nom d'usuari i/o contrasenya incorrectes.</font></font>");
    e->show();
}

}

// funció utilitzada per a tancar la finestra d'autenticació, s'executa quan l'usuari prem el botó de cancel·lar
void Autenticacio::cancel()
{
    this->close();
}

```

A continuació mostrem la secció del codi que s'encarrega d'administrar les diferents possibilitats dins de la finestra principal de l'aplicació. Aquest codi es pot trobar en el fitxer *principal.ui.h*.

```

#include <qapplication.h>
#include <qlabel.h>
#include <qlineedit.h>
#include <qstring.h>
#include <qfiledialog.h>
#include <qstatusbar.h>
#include <qpushbutton.h>
#include "canvipass.h"
#include "altacirurgia.h"
#include "baixacirurgia.h"
#include "about.h"
#include "continguts.h"
#include "sortida.h"
#include <iostream>
#include <fstream>
#include <string.h>

using namespace std;
QString filename="";

// funció que en un futur servirà per a carregar els diferents models prostàtics de l'aplicació.
void Simpros::fileOpen()

```

```

{
    filename = QFileDialog::getOpenFileName( QString::null, QString::null, this);
    if ( !filename.isEmpty() )
        load( filename );
    else
        statusBar()->message( "Loading aborted", 2000 );
}

//funció auxiliar per a la càrrega de models
void Simpros::load( const QString &fileName )
{
    QFile f( fileName );
    if ( !f.open( IO_ReadOnly ) )
        return;

    QTextStream ts( &f );

    setCaption( fileName );
    statusBar()->message( "Loaded document " + fileName, 2000 );
}

//funció que en un futur servirà per a guardar els models prostàtics
void Simpros::fileSave()
{
    if ( filename.isEmpty() ) {
        fileSaveAs();
        return;
    }

    QFile f( filename );
    if ( !f.open( IO_WriteOnly ) ) {
        statusBar()->message( QString("Could not write to %1").arg(filename), 2000 );
        return;
    }

    f.close();

    setCaption( filename );
    statusBar()->message( QString( "File %1 saved" ).arg( filename ), 2000 );
}

//funció que en un futur servirà per a guardar models amb un nom determinat
void Simpros::fileSaveAs()
{
    QString fn = QFileDialog::getSaveFileName( QString::null, QString::null, this );
    if ( !fn.isEmpty() ) {
        filename = fn;
        fileSave();
    }

    else {
        statusBar()->message( "Saving aborted", 2000 );
    }
}

//funció per a tancar l'aplicació
void Simpros::fileExit()
{
    this->close();
}

//funció per a mostrar els continguts bàsics del funcionament de l'aplicació
void Simpros::helpContents()
{
    Continguts *co=new Continguts();
    co->show();
}

//funció per a mostrar informació relativa a la versió i el creador de l'aplicació
void Simpros::helpAbout()
{
    About *ab = new About();
    ab->show();
}

```

```

//funció per a canviar la contrasenya dels cirurgians
void Simpros::canviPassCir()
{
    Canvipass *c= new Canvipass();
    c->show();
}

//funció per a canviar la contrasenya de l'administrador
void Simpros::canviPassAdm()
{
    Canvipass *c= new Canvipass();
    c->textUsuari->hide();
    c->lineNom->hide();
    c->setCaption("Canviar password Administrador");
    c->textLabel1->setText("<font size='1'\><font size='1'\>Introdueix els camps necessaris per a realitzar el  
canvi de contrasenya d'un usuari administrador.</font></font>");
    c->show();
}

//funció per a donar d'alta a nous cirurgians
void Simpros::AltaCir()
{
    AltaCirurgia *a=new AltaCirurgia();
    a->show();
}

//funció per a donar de baixa a cirurgians
void Simpros::BaixaCir()
{
    BaixaCirurgia *b=new BaixaCirurgia();
    b->show();
}

```

La resta del codi de l'aplicació segueix l'estructura mostrada anteriorment i es troba en el cd de l'aplicació Simpros.

5 Manual d'usuari

En aquest apartat es definirà un manual d'usuari per tal de que els usuaris de l'aplicació coneguin en detall cadascuna de les funcionalitats i la manera de poder realitzar-les. Per tant, aquest manual serà bàsic pels usuaris no avançats en l'aplicació o pels que requereixin recordar alguns aspectes funcionals. A continuació detallarem el manual d'usuari pels dos paquets de treball de l'aplicació: Interaccions amb model i Gestió d'usuaris i fitxers.

5.1 Interaccions amb model

En aquest paquet de treball es defineixen les instruccions per poder realitzar la part de simulació amb el model a partir del *haptic Phantom Desktop*.

Les funcionalitats de **moviment** o **gir** del resectoscopi es realitzen a partir de variacions en les senyals d'entrada del *haptic*. Aquestes senyals són sis i corresponen als sis graus de llibertat que conté. Per tant, a partir de combinacions de variacions d'aquestes senyals es poden obtenir els girs o moviments requerits durant la simulació. Aquestes sis senyals d'entrada són:

1. **Component de l'eix x**
2. **Component de l'eix y**
3. **Component de l'eix z**
4. **Angle amb recolzament x:** per a pivotar el resectoscopi cap a la dreta o esquerra
5. **Angle amb recolzament y:** per a pivotar el resectoscopi cap amunt o cap avall
6. **Angle rotatori z:** pels girs horaris/antihoraris

La funcionalitat d'**activar el pas de corrent de nivell de tall** es realitza prement sobre el botó que incorpora el *haptic*. Contràriament, per a **desactivar el pas de corrent** de l'anella de tall és necessari tornar a prémer sobre el botó. La combinació del pas de corrent juntament amb els moviments o girs explicats anteriorment permeten realitzar les operacions de **tall** de la simulació. L'**activació del pas de corrent de nivell de cauterització** es realitza prement la tecla **c** del teclat. De la mateixa manera que en el cas anterior, la desactivació és possible si es torna a prémer sobre la tecla **c**.

La funcionalitat restants de la interacció amb el model es realitzen manteniment premudes les següents tecles del teclat:

- **a: Injectar aigua**
- **x: Estendre l'anella de tall**
- **z: Retractar l'anella de tall**

5.2 Gestió d'usuaris i fitxers

En aquest paquet de treball es defineixen les funcionalitats que requereixen interacció entre l'usuari i la interfície. A continuació definim les instruccions per a cadascuna d'aquestes funcionalitats i les regles de validació aplicades:

- **Carregar model prostàtic:** aquesta funcionalitat serveix per carregar un model prostàtic des d'un fitxer. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació. L'operació es realitza premen a sobre del component *Carregar model* dins del menú *Arxiu* de la finestra principal.
- **Guardar model prostàtic:** aquesta funcionalitat serveix per guardar un model prostàtic dins d'un fitxer. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació. L'operació es realitza premen a sobre del component *Guardar model* dins del menú *Arxiu* de la finestra principal.
- **Sortir:** aquesta funcionalitat serveix per sortir de l'aplicació. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació. L'operació es realitza premen a sobre del component *Sortir* dins del menú *Arxiu* de la finestra principal.
- **Autenticació:** aquesta funcionalitat serveix per autenticar els usuaris a l'inici. L'operació es realitza en la finestra d'autenticació i requereix la introducció de dos camps amb les següents característiques i la posterior acceptació:

- Nom d'usuari: Necessàriament ha de ser el nom d'un usuari existent en l'aplicació.
 - Password: Necessàriament ha de ser la contrasenya correcta de l'usuari introduït a sobre.
- **Alta cirurgia:** aquesta funcionalitat serveix per donar d'alta a nous usuaris en l'aplicació. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació i que l'usuari autenticat sigui l'administrador del sistema. L'operació es realitza premen a sobre del component *Alta cirurgia* dins del menú *Gestió Usuaris* de la finestra principal i requereix la introducció de dos camps amb les següents característiques i la posterior acceptació:
- Nom d'usuari: Necessàriament ha de ser el nom d'un usuari inexistent en l'aplicació.
 - Password: Obligatòriament el número de caràcters de la contrasenya ha d'estar comprés entre 3 i 8.
- **Baixa cirurgia:** aquesta funcionalitat serveix per donar de baixa a usuaris existents en l'aplicació. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació i que l'usuari autenticat sigui l'administrador del sistema. L'operació es realitza premen a sobre del component *Baixa cirurgia* dins del menú *Gestió Usuaris* de la finestra principal i requereix la introducció d'un camp amb les següents característiques i la posterior acceptació:
- Nom d'usuari: Necessàriament ha de ser el nom d'un usuari existent en l'aplicació.
- **Canvi password cirurgia:** aquesta funcionalitat serveix per modificar la contrasenya dels usuaris de tipus cirurgians de l'aplicació. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació. L'operació es realitza premen a sobre del component *Canvi password cirurgia* dins del menú *Gestió Usuaris* de la

finestra principal i requereix la introducció de quatre camps amb les següents característiques i la posterior acceptació:

- Nom d'usuari: Necessàriament ha de ser el nom d'un usuari existent en l'aplicació.
- Password actual: Necessàriament ha de ser la contrasenya correcta de l'usuari introduït a sobre. Aquest camp només apareix si el canvi de contrasenya el realitza el propi cirurgià.
- Nova password: Obligatòriament el número de caràcters de la contrasenya ha d'estar comprés entre 3 i 8.
- Verificació nova password: Necessàriament ha de ser la mateixa contrasenya que la introduïda en *Nova password*.

- **Canvi password administrador:** aquesta funcionalitat serveix per modificar la contrasenya dels usuaris de tipus cirurgians de l'aplicació. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació i que l'usuari autenticat sigui l'administrador del sistema. L'operació es realitza premen a sobre del component *Canvi password administrador* dins del menú *Gestió Usuaris* de la finestra principal i requereix la introducció de tres camps amb les següents característiques i la posterior acceptació:

- Password actual: Necessàriament ha de ser la contrasenya correcta de l'administrador del sistema.
- Nova password: Obligatòriament el número de caràcters de la contrasenya ha d'estar comprés entre 3 i 8.
- Verificació nova password: Necessàriament ha de ser la mateixa contrasenya que la introduïda en *Nova password*.

- **Continguts d'ajuda:** aquesta funcionalitat serveix per consultar l'ajuda de l'aplicació de manera esquemàtica i sintetitzada. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació. L'operació es realitza premen a sobre del component *Continguts* dins del menú *Ajuda* de la finestra principal.

- **Informació de l'aplicació:** aquesta funcionalitat serveix per consultar informació referent al producte. Es mostren dades tals com la versió i l'autor. Un requisit previ a la realització d'aquesta operació es haver fet satisfactòriament l'autenticació. L'operació es realitza premen a sobre del component *About* dins del menú *Ajuda* de la finestra principal.

6 Anàlisi de la planificació i valoració econòmica

En aquest apartat es realitza una comparativa entre la planificació dissenyada a l'inici del projecte i la planificació definitiva un cop acabat el projecte. Per altra banda, es realitza una valoració econòmica del projecte segons els rols necessaris en el projecte i les hores imputades.

6.1 Anàlisi de la planificació

Seguint la metodologia **Rational Unified Process (RUP)** aplicada al conjunt del projecte, la planificació s'estructura en dues fases: la *iniciació* i l'*elaboració*.

La primera fase o fase d'iniciació, ha permès d'establir els límits del projecte, fer la planificació, fer una gestió dels riscos que poden aparèixer en el projecte, obtenir informació sobre les operacions de pròstata via uretral i desenvolupar els experiments necessaris en el simulador actual per tal d'habituar-se al seu funcionament.

En la fase d'elaboració hem realitzat dues iteracions. En la primera hem produït un llistat dels requisits no funcionals del sistema i hem realitzat totes les tasques, els documents i els diagrames propis de l'especificació del sistema seguint la metodologia mencionada anteriorment. Posteriorment refinarem tota aquesta anàlisi. En la segona hem dissenyat les tres capes de les que consta el sistema, hem implementat la interfície d'usuari i hem realitzat la valoració econòmica del projecte.

Les tasques del projecte es varen iniciar a mitjans de juliol de 2007. La productivitat del mes d'agost es considera inferior a la inicialment prevista. La dedicació al desenvolupament del projecte es va fixar en un valor aproximat de tres hores diàries però finalment el valor promig ha estat de quatre hores diàries inclosos molts caps de setmana que en el diagrama de Gantt es veuen reflectits en hores afegides als dies laborables. Els dies que apareixen en la planificació responen a aquesta càrrega de treball.

A continuació descriurem un anàlisi comparatiu entre la planificació dissenyada a l'inici del projecte i la planificació definitiva un cop acabat el projecte per a la fase d'incepció, per a la primera iteració de la fase d'elaboració i per a la segona iteració de la fase d'elaboració.

La planificació estimada inicialment per a la fase d'incepció ha estat equivalent a la planificació real. Això ha estat possible gràcies a que la quantitat de tasques per a aquesta fase ha estat molt reduïda i a que les seves tasques tenien finalitats molt puntuals en un curt espai de temps. En la Figura 6.1 s'il·lustra la planificació per a la fase d'incepció.



Figura 6.1: Planificació per a la fase d'incepció

Per altra banda, l'estimació inicial per a la fase d'elaboració no ha estat gens encertada. La realitat en quant a aquesta fase ha estat molt diferent. Concretament, en la primera iteració d'aquesta fase la duració real ha estat tretze dies superior a l'estimació inicial. És important esmentar que aquest increment d'hores ha estat produït principalment per a l'augment d'hores de treball en la realització dels diagrames de seqüència de l'especificació i per l'aparició de dues tasques que no estaven pròpiament definides a l'inici: descripció dels casos d'ús i elaboració dels contractes de les operacions de sistema. La Figura 6.2 il·lustra la planificació estimada a l'inici del projecte per a la primera iteració de la fase d'elaboració, la Figura 6.3 il·lustra la planificació real de la primera iteració de la fase d'elaboració en format de dates i la Figura 6.4 il·lustra la planificació real de la primera iteració de la fase d'elaboració en format de diagrama de Gantt.

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1	Elaboració	31 días	lun 03/09/07	lun 15/10/07	
2	Iteració1	31 días	lun 03/09/07	lun 15/10/07	
3	Anàlisi i Disseny	31 días	lun 03/09/07	lun 15/10/07	
4	Listar requisitos no funcionals	1 día	lun 03/09/07	lun 03/09/07	
5	Diagrama de blocs d'alt nivell	1 día	lun 03/09/07	lun 03/09/07	
6	Use Case Model	5 días	lun 03/09/07	vie 07/09/07	
7	Analitzar casos d'ús més significatius	1 día	lun 10/09/07	lun 10/09/07	6
8	Diagrames d'estats	4 días	lun 03/09/07	jue 06/09/07	
9	Diagrames de seqüència	2 días	lun 10/09/07	mar 11/09/07	6
10	Refinar diagrames d'anàlisi	7 días	mié 12/09/07	jue 20/09/07	9
11	Supplementary Specifications	4 días	vie 21/09/07	mié 26/09/07	10
12	Software Architecture Document	5 días	jue 27/09/07	mié 03/10/07	11
13	Analysis Model	7 días	jue 04/10/07	vie 12/10/07	12
14	Glossary	1 día	lun 15/10/07	lun 15/10/07	13

Figura 6.2: Estimació de la planificació per a la primera iteració de la fase d'elaboració

	Nombre de tarea	Duration	Start	Finish	Predecessors
1	Elaboració	44 days	Mon 03/09/07	Thu 01/11/07	
2	Iteració1	44 days	Mon 03/09/07	Thu 01/11/07	
3	Anàlisi i Disseny	44 days	Mon 03/09/07	Thu 01/11/07	
4	Listar requisitos no funcionals	1 day	Mon 03/09/07	Mon 03/09/07	
5	Diagrama de blocs d'alt nivell	1 day	Mon 03/09/07	Mon 03/09/07	
6	Use Case Model	5 days	Tue 04/09/07	Mon 10/09/07	5
7	Descripció casos d'ús	7 days	Tue 11/09/07	Wed 19/09/07	6
8	Diagrames d'estats	4 days	Tue 04/09/07	Fri 07/09/07	5
9	Diagrames de seqüència	10 days	Tue 11/09/07	Mon 24/09/07	6
10	Refinar diagrames d'anàlisi	4 days	Tue 25/09/07	Fri 28/09/07	9
11	Contractes de les operacions	7 days	Mon 01/10/07	Tue 09/10/07	10
12	Supplementary Specifications	5 days	Wed 10/10/07	Tue 16/10/07	11
13	Software Architecture Document	5 days	Wed 17/10/07	Tue 23/10/07	12
14	Analysis Model	7 days	Wed 24/10/07	Thu 01/11/07	13
15	Glossary	1 day	Mon 01/11/07	Mon 01/11/07	10

Figura 6.3: Planificació real per a la primera iteració de la fase d'elaboració

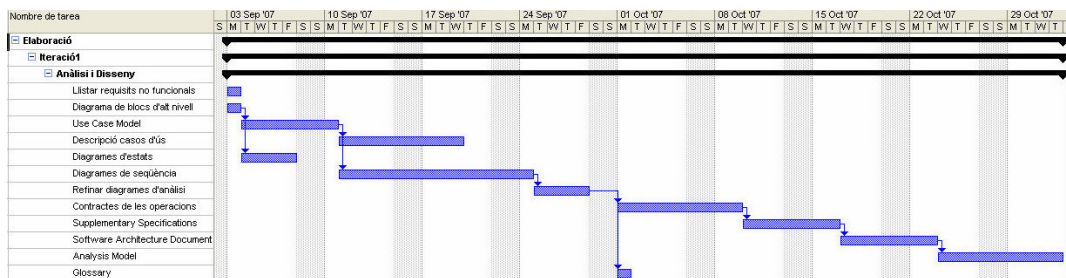


Figura 6.4: Planificació real per a la primera iteració de la fase d'elaboració

En quant a la segona iteració de la fase d'elaboració, el número de dies treballats s'ha vist incrementat en cinc dies respecte a l'estimació inicial. Això ha estat així degut a que algunes tasques requerien treball previ i perquè també s'ha afegit la implementació de la interfície d'usuari com a nova tasca a realitzar. A més, hi ha dues tasques principals que s'han quedat enrederides respecte a l'estimació: el *design model* i la realització dels diagrames de seqüència del disseny del sistema. La Figura 6.5 il·lustra la planificació estimada a l'inici del projecte per a

la segona iteració de la fase d'elaboració, la Figura 6.6 il·lustra la planificació real de la segona iteració de la fase d'elaboració en format de dates i la Figura 6.7 il·lustra la planificació real de la segona iteració de la fase d'elaboració en format de diagrama de Gantt.

		Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1		Elaboració	41 días	mié 26/09/07	mié 21/11/07	
2		Iteració2	41 días	mié 26/09/07	mié 21/11/07	
3		Anàlisi i Disseny	32 días	mié 26/09/07	jue 08/11/07	
4		Design Model	5 días	mié 26/09/07	mar 02/10/07	
5		Software Architecture Document	7 días	mié 03/10/07	jue 11/10/07	4
6		StoryBoards	1 día	mié 26/09/07	mié 26/09/07	
7		Navigation Map	1 día	jue 27/09/07	jue 27/09/07	6
8		Disseny capa presentacio	7 días	mié 03/10/07	jue 11/10/07	4,7
9		Disseny capa de domini	10 días	vie 12/10/07	jue 25/10/07	8
10		Disseny capa de gestió de dades	4 días	vie 26/10/07	mié 31/10/07	9
11		Diagrames de seqüència	5 días	jue 01/11/07	mié 07/11/07	10,8,9
12		Glossary	1 día	jue 08/11/07	jue 08/11/07	11
13		Gestió de projectes	9 días	vie 09/11/07	mié 21/11/07	
14		Software Development Plan	3 días	vie 09/11/07	mar 13/11/07	3
15		Planificació real	2 días	mié 14/11/07	jue 15/11/07	14
16		Valoració econòmica del projecte	4 días	vie 16/11/07	mié 21/11/07	15

Figura 6.5: Estimació de la planificació per a la segona iteració de la fase d'elaboració

		Nombre de tarea	Duration	Start	Finish	Predecessors
1		Elaboració	46 days	Fri 02/11/07	Tue 08/01/08	
2		Iteració2	46 days	Fri 02/11/07	Tue 08/01/08	
3		Anàlisi i Disseny	34 days	Fri 02/11/07	Wed 19/12/07	
4		Arquitectura física del sistema	5 days	Fri 02/11/07	Thu 08/11/07	
5		Design Model	8 days	Fri 09/11/07	Tue 20/11/07	4
6		Disseny capa de gestió de dades	2 days	Wed 21/11/07	Thu 22/11/07	4,5
7		StoryBoards	1 day	Fri 09/11/07	Fri 09/11/07	4
8		Navigation Map	3 days	Mon 12/11/07	Wed 14/11/07	7
9		Diagrames de seqüència del disseny	9 days	Wed 21/11/07	Mon 03/12/07	5
10		Disseny capa presentacio	6 days	Tue 04/12/07	Tue 11/12/07	8,9
11		Disseny capa de domini	6 days	Tue 04/12/07	Tue 11/12/07	5,9,6
12		Software Architecture Document	5 days	Wed 12/12/07	Tue 18/12/07	6,11,10
13		Glossary	1 day	Wed 19/12/07	Wed 19/12/07	12
14		Implementació	11 days	Wed 12/12/07	Fri 28/12/07	
15		Interfície d'usuari amb Qt	9 days	Wed 12/12/07	Mon 24/12/07	11
16		Manual d'usuari	2 days	Thu 27/12/07	Fri 28/12/07	15
17		Gestió de projectes	7 days	Mon 31/12/07	Tue 08/01/08	
18		Anàlisi de la planificació	2 days	Mon 31/12/07	Tue 01/01/08	3,14
19		Valoració econòmica del projecte	2 days	Wed 02/01/08	Thu 03/01/08	18
20		Software Development Plan	3 days	Fri 04/01/08	Tue 08/01/08	19

Figura 6.6: Planificació real per a la segona iteració de la fase d'elaboració

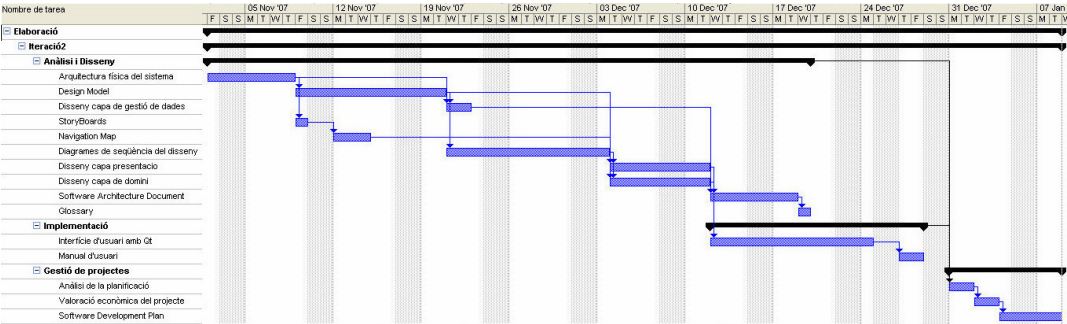


Figura 6.7: Planificació real per a la segona iteració de la fase d'elaboració

6.2 Valoració econòmica

Per tal de valorar econòmicament el projecte desglossarem les hores dedicades a cadascuna de les seves fases. Recordem que tal i com hem esmentat abans, la dedicació ha estat de quatre hores diàries de mitja. Les hores treballades corresponen a les hores reals que he utilitzat per realitzar les tasques tenint en compte que alguns dies he treballat més de quatre hores i també alguns caps de setmana. En quant als dies de treball només estic contant en un primer moment els laborables.

Fase	Dies de treball (Elapsed Time)	Hores treballades
Incepció	12	72
Elaboració: primera iteració	43	228
Elaboració: segona iteració	39	256
Total	94	556

El nombre d'hores totals del projecte s'eleva a **556**.

La següent tasca a realitzar és la valoració econòmica tenint en compte que el projecte l'hagués realitzar alguna empresa. Aquesta tasca s'ha estimat amb la mitja de les tarifes que cobren cadascun dels rols en el mercat actual. La següent taula mostra l'assignació de tasques als diferents rols, el cost que això suposaria i el preu global del projecte:

Tasques	Perfil	Hores	Tarifa (€ / hora)	Preu (€)
- Planificació i gestió del projecte - Definició de requeriments	Cap de projecte	76	60	4560
- Anàlisi i disseny del sistema	Analista	412	40	16480
- Implementació de la interfície d'usuari i - Pràctiques amb el simulador actual	Programador	68	30	2040
Total		556		23080

7 Conclusions

En aquest apartat es valoren les conclusions personals que s'han tret de la realització d'aquest projecte. A més, es descriu el treball que fa falta realitzar en un futur per a que el simulador prostàtic arribi a ser una realitat per als cirurgians que estan en fase d'entrenament.

7.1 Conclusions personals

Aquest projecte m'ha suposat adquirir una maduració molt important i un aprenentatge molt intens en quant a la realització de projectes informàtics. Això vol dir, que tot el que he anat realitzant em servirà força de cara al meu futur en el món laboral. És important esmentar que aquest ha estat el primer gran projecte que he realitzat individualment i en el qual he tingut que planificar satisfactòriament la càrrega de treball corresponent al projecte.

Concretament, he après moltes coses en un tema tan important com són les operacions de pròstata, que vulguis o no, acostuma a afectar a la gran majoria d'homes algun cop a la vida. Conèixer en profunditat com funcionen aquestes operacions em sembla una cosa molt interessant. Aquesta va ser una de les raons per les quals vaig triar aquest projecte. Per altra banda, he adquirit uns bons coneixements sobre la forma de treballar amb una metodologia com és *Rational Unified Process (RUP)*, que ja l'havia utilitzat en la realització del projecte d'Enginyeria del Software.

Considero que és molt important que aquest projecte es possi en pràctica en els hospitals estatals ja que és bàsic que els cirurgians tinguin una eina efectiva amb la qual fer les pràctiques pertinents abans d'adquirir els coneixements suficients com per operar a pacients reals. L'adaptació a l'eina per part dels cirurgians és un procés probablement difícil però amb el temps les simulacions d'aquest tipus d'operacions seran una rutina diària per a molts d'ells. Crec que avui en dia molts cirurgians no adquireixen els suficients coneixements ni experiència com per realitzar operacions de pròstata de manera admissible per als pacients. Això és degut a que els no experimentats simplement assisteixen a operacions reals i visualitzen les tècniques de l'operació.

7.2 Treball futur

Donat que aquest projecte és la meitat d'un macro projecte en el qual l'altre meitat consisteix en la implementació de tota la capa de domini del simulador, es requereix que aquesta segona part s'implementi aviat. La persona indicada per a implementar aquesta segona part haurà de tenir grans coneixements d'OpenGL i de geometria. Continuant amb la metodologia *Rational Unified Process (RUP)*, falten per a realitzar les fases de construcció i transició. La fase de construcció està formada bàsicament per la implementació de l'aplicació i per les proves necessàries per garantir que el simulador funciona de manera adequada. La fase de transició consisteix en la instal·lació del simulador en els hospitals i en l'ensenyament als cirurgians de com han d'entrenar-se amb el simulador. El que és molt interessant és que la persona que es dediqui a realitzar la segona part del macro projecte tingui molt en compte el que s'ha especificat i dissenyat durant el transcurs del meu projecte. Per tant, té una base sòlida de treball que li servirà molt.

8 Agraïments

Vull agrair a en Robert Joan Arinyo per a haver confiat en mi en tot moment durant el transcurs del projecte i per tenir una metodologia de treball amb l'estudiant molt adequada. També vull agrair a una altre peça important d'aquest projecte: Josep Vilaplana. Gràcies a ell vaig poder fer les simulacions necessàries sobre l'aplicació actual.

Vull agrair també a la meva novia Verònica per recolzar-me durant les dures etapes d'aquest projecte, als meus pares que sempre s'interessen per mi en tot el que faig i a la meva germana Cristina. A més, una menció especial per als meus amics que els he deixat una mica de banda amb tota la càrrega de treball que comporta el projecte.

9 Bibliografia i referències

La bibliografia i les referències electròniques són una part important en la realització de projecte ja que mostren les fonts d'informació utilitzades i, per tant, són referències que podran ser consultades en un futur.

9.1 Bibliografia

A continuació indico la bibliografia utilitzada durant la realització del meu projecte:

- D. Costal, X. Franch, M. R. Sancho, E. Teniente. *Enginyeria del software Especificació. Especificació de sistemes orientats a objectes amb la notació UML*. Tercera edició. Edicions UPC, febrer del 2005.
- Institut d'estudis catalans. *Diccionari de llengua catalana*. Tercera edició. Edicions 62, Editorial Moll, 1995.
- Transparències d'ES2 (Enginyeria del Software 2)
- R. Joan-Arinyo, J. Vilaplana. *Interactive Cutting in Voxel-based Objects*. 2006

9.2 Referències electròniques

A continuació indico les referències electròniques utilitzades durant la realització del meu projecte:

- <http://doc.trolltech.com/3.3/designer-manual.html>
- <http://www.sensable.com/haptic-phantom-desktop.htm>
- <http://www.kdehispano.org/node/1069>
- <http://dac.escet.urjc.es/rvmaster/asignaturas/MCDH/claseClasificacionDispositivosHapticosbn.pdf>
- <http://www-306.ibm.com/software/awdtools/rup/>

10 Annexos

Els annexos de la memòria escrita contenen el glossari de termes importants enllaçats amb les seves definicions corresponents.

10.1 Glossari

El glossari descrit a continuació conté la definició de termes importants de la memòria ordenats alfabèticament.

- **Administrador:** Usuari que té la responsabilitat sobre el sistema i que té permissos especials.
- **Anella de tall:** Extremitat del resectoscopi que serveix per a realitzar talls durant les operacions.
- **Cànula:** Tub curt que forma part del resectoscopi i s'emplea en diferents operacions quirúrgiques.
- **Cirurgià:** Usuari encarregat de realitzar les simulacions en el sistema com a part del seu entrenament previ a les operacions reals de pròstata.
- **Haptic:** Conjunt de interfícies tecnològiques que interaccionen amb el ser humà mitjançant el sentit del tacte i que permeten fer simulacions d'operacions quirúrgiques o de vol.
- **Interacció:** Acció que s'exerceix recíprocament entre els usuaris del sistema i el simulador prostàtic.
- **Pròstata:** Glàndula de secreció externa, pròpia del mascle, que secreta un líquid viscos i opalescent.
- **Rational Unified Process (RUP):** és un procés de desenvolupament de software i juntament amb el *Unified Modified Language (UML)* constitueix la metodologia estàndard més utilitzada per a l'anàlisi, implementació i documentació de sistemes orientats a objectes.
- **Resectoscopi:** Bisturí utilitzat com a eina principal pels cirurgians en les operacions de pròstata.

- **Simpros:** Nom del simulador d'operacions de prostàtes dissenyat en aquest projecte.
- **Usuari:** Usuari genèric de l'aplicació que pot ser un cirurgià o l'administrador del sistema.
- **Varetes:** Extremitats de ferro que sostenen l'anella de tall i que formen part del resectoscopi.
- **Vòxel:** Unitat mínima de l'aplicació que equival a un píxel tridimensional.